RI-5H6
393

SCB-1020

RS-232 SYNCHRONOUS / ASYNCHRONOUS BOARD

for IBM PC/AT

INTERFACE CARDS FOR IBM PC/AT AND PS/2

TEL.: (216) 434-3154
FAX: (216) 434-1409
TELEX: 5101012726

QUA TECH, INC.
478 East Exchange St.
Akron, OH 44304

WARRANTY INFORMATION

Qua Tech Inc. warrants the SCB-1020 to be free of defects for one (1) year from the date of purchase. Qua Tech Inc. will repair or replace any board that fails to perform under normal operating conditions and in accordance with the procedures outlined in this document during the warranty period. Any damage that results from improper installation, operation, or general misuse voids all warranty rights.

Although every attempt has been made to guarantee the accuracy of this manual, Qua Tech Inc. assumes no liability for damages resulting from errors in this document. Qua Tech Inc. reserves the right to edit or append to this document at any time without notice.

Please complete the following information and retain for your records. Have this information available when requesting warranty service.

DATE OF PURCHASE: _____

MODEL NUMBER: SCB-1020

PRODUCT DESCRIPTION: SINGLE CHANNEL EIA-232 MULTI-PROTOCOL COMMUNICATINS BOARD

SERIAL NUMBER: _____

IBM™ is a registered trademarks of International Business Machines.

TABLE OF CONTENTS

WARRANTY.....i

TABLE OF CONTENTS.....ii

LIST OF FIGURES.....iv

I. INTRODUCTION.....1

II. GENERAL INFORMATION.....2
A. SCB-1020 PARTS LAYOUT.....2
B. SPECIFICATIONS.....3

III. GETTING STARTED.....4
A. ADDRESS SELECTION.....4
(1) BASE ADDRESSING.....4
(2) FACTORY ADDRESS SETTINGS.....6
B. CONFIGURATIONS.....7
(1) INTERRUPT LEVEL SELECT.-J1.....7
(2) DMA SELECT-J2.....7
(3) INTERRUPT MODE SELECT-J3.....8
(4) EIA-232 DTE\DCE SELECT-J4.....8
(5) INTERRUPT SOURCE AND RTS/CTS CONTROL SELECT-J5.....10
(6) TRANSMIT CLOCK SOURCE SELECT-J6.....11
(7) RECEIVE CLOCK SOURCE SELECT-J7.....12
(8) -SYNC/DSR SELECT-J8.....12
(9) JUMPER SUMMARY TABLE.....13
(10) FACTORY JUMPER CONFIGURATIONS.....15
C. EXTERNAL CONNECTIONS.....16

IV. PROGRAMMING THE SCB-1020.....19
A. OPERATIONAL DESCRIPTION.....19
(1) 8254 PROGRAMMABLE COUNTER/TIMER.....19
(2) μPD7201A MULTIPLE PROTOCOL SERIAL CONTROLLER.....21
B. PROGRAMMING EXAMPLES.....22
(1) EXAMPLE1.....22
(2) EXAMPLE2.....22
(3) EXAMPLE3.....23
(4) EXAMPLE4.....25

V. EIA STANDARD SUPPORTED BY THE SCB-1020.....27
A. EIA-232 STANDARD.....27

VI. DETAILED INFORMATION ON THE 8254.....28

VII. DETAILED INFORMATION ON THE μPD7201A.....31
A. CONTROL REGISTER 0.....31
B. CONTROL REGISTER 1.....31
C. CONTROL REGISTER 2.....31
D. CONTROL REGISTER 2B.....31
E. CONTROL REGISTER 3.....32
F. CONTROL REGISTER 4.....32
G. CONTROL REGISTER 5.....32
H. CONTROL REGISTER 6.....32
I. CONTROL REGISTER 7.....32

J. STATUS REGISTER 0.....32
K. STATUS REGISTER 1.....32
L. STATUS REGISTER 2B.....32
M. STATUS REGISTER 3.....33
N. STATUS REGISTER 4.....33

VIII. ANSWERS TO COMMONLY ASKED QUESTIONS.....34

IX. NEC'S DATA SPECIFICATIONS ON THE μPD7201.....36

LIST OF FIGURES

FIGURE 1 SCB-1020 PARTS LAYOUT.....2

FIGURE 2 EIA SPECIFICATIONS.....3

FIGURE 3 8254 AND μPD7201A INTERNAL REGISTERS.....5

FIGURE 4 HARDWARE FOR INTERRUPT SHARING CAPABILITIES..8

FIGURE 5 DTE SIGNAL DIRECTION.....9

FIGURE 6 DCE SIGNAL DIRECTION.....9

FIGURE 7 EIA-232 SIGNALS AND DIRECTIONS ON D-25.....10

FIGURE 8 JUMPER SUMMARY TABLE.....13

FIGURE 9 FACTORY JUMPER CONFIGURATIONS.....15

FIGURE 10 D-25 PINOUT.....16

FIGURE 11 μPD7201A CONTROL AND STATUS REGISTERS.....21

FIGURE 12 PIN-OUT OF MOTOROLA'S MC1488 AND MC1489....27

FIGURE 13 PIN-OUT OF INTEL'S 8254.....28

FIGURE 14 MODE CONTROL FORMAT FOR THE 8254.....28

I. INTRODUCTION

The SCB-1020 is a single channel synchronous communication board which supports EIA-232 communication. The SCB-1020 incorporates the NEC μPD7201A multi-protocol serial communication controller to allow data transfers up to 1Mbaud. Although the μPD7201A is capable of transferring data up to 1Mbaud, there are limitations due to EIA-232 physical specifications. Please refer to FIGURE 2 of page 4 for proper operations using EIA Serial Bus Specifications. Direct Memory Access (DMA) is also supported to increase data throughput. The SCB-1020 can handle asynchronous and synchronous byte oriented protocols such as IBM Bisync, as well as synchronous bit oriented protocols such as HDLC and SDLC. The SCB-1020 is a 7.5 inch card which occupies only one expansion slot. Switch selectable addressing allows the board to be installed in the computer without conflicting with already existing devices. The EIA-232 signals are available via a D-25 connector which provides excellent shielding against environmental noise. Versatile interrupt circuitry allows the SCB-1020 access to IRQ2-IRQ7. The SCB-1020 may also share any of these interrupts with another device such as Qua Tech's DS-101, DS-201, or DS-225. DMA channels 1 and 3 are available to allow data transfers up to 200 Kbyte/sec. A single channel may be used to transfer and receive information or both DMA channels 1 and 3 may be used, one to transmit data and one to receive data. All DMA configurations are jumper selectable. Data transfer rates are established by the transmit and receive clock inputs of the μPD7201A. These input clocks may be from an outside source or provided by an on board programmable timer.

In the following pages, a step by step approach is taken to guide the user in configuring, programming and troubleshooting the SCB-1020. It should be no time at all before the SCB-1020 is up and running.

II. GENERAL INFORMATION

A. SCB-1020 PARTS LAYOUT

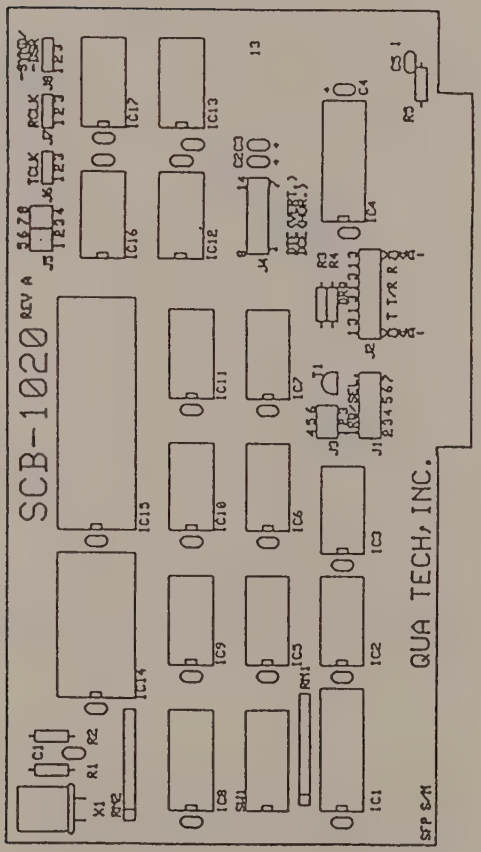


FIGURE 1

B. SPECIFICATIONS

BUS: IBM PC/XT

DIMENSIONS: 7.5" X 3.9" X .062"

POWER: 400 mA @ +5 VOLTS
26 mA @ +12 VOLTS
23 mA @ -12 VOLTS

ADDRESS: SWITCH SELECTABLE

BUS LOAD: MAXIMUM OF ONE LSTTL
LOAD PER BUS LINE

The following table is the EIA specifications for various serial busses.

EIA SPECIFICATIONS

Serial Bus Specifications			
Parameter	RS-232-C	RS-422-A	RS-485 (Bitbus)
Mode of operation	Single-ended	Differential	Differential
Number of drivers and receivers allowed on a line	1 driver 1 receiver	1 driver 1 receiver	32 drivers 32 receivers
Maximum cable length	50 ft	4,000 ft	4,000 ft
Maximum baud rate	20 kbaud	10,000 kbaud	10,000 kbaud
Maximum common mode voltage	±25 V	+6 to −0.25 V	+12 to −7 V
Driver output	±5 V min ±5 V max	±2 V min	±1.5 V min
Driver load	3k to 7k ohms	100 ohms	60 ohms
Receiver input resistance	3k to 7k ohms	4k ohms	4k ohms
Receiver sensitivity	±3,000 mV	±200 mV	±200 mV

FIGURE 2

III. GETTING STARTED

Thank you for purchasing the SCB-1020 synchronous communication board. We hope that it brings you as many rewarding results as it did us in designing it.

The initial board setup can be broken down into three areas: address selection, jumper configurations, and connections from the SCB-1020 to any external device.

A. ADDRESS SELECTION

(1) BASE ADDRESSING

The address of the SCB-1020 is user selectable by the eight position switch SW1. This switch allows the SCB-1020 to reside anywhere in the I/O port address space from 000H to 3F8H on eight byte boundaries. An eight byte boundary simply implies that the SCB-1020 needs eight addresses to access the internal registers of the 8254 programmable timer/counter and the μPD7201A multiprotocol serial controller (refer to FIGURE 3 for a complete list of the 8254's and the μPD7201A's internal registers). For example, if the base address of the board is set to 300H, then the SCB-1020's internal registers are accessed by address 300H to 307H. Although the board may use any address from 000H to 3F8H as dictated by SW1, the allowable I/O spaces for the IBM PC/XT/AT are as follows:

210H - 277H
280H - 2F7H
300H - 377H

The user may select any address outside these ranges but within 000H to 3F8H provided no other device is installed at that address.

Any board installed in the computer through one of the expansion slots is defined as I/O mapped. An I/O mapped board uses only sixteen of the twenty address bits provided by the computer. The SCB-1020 uses only ten of these sixteen address bits. The eight position switch allows user selection of address bits A3-A9. These seven address bits form the base address of the SCB-1020. Bits A0, A1, and A2 are not user selectable. Address bit A2 is used to select either the 8254 programmable timer/counter (A2=0) or the μPD7201A multiprotocol serial controller (A2=1). Address bits A0 and A1 select the four locations within the 8254 and μPD7201A. The three examples that follow illustrate how the address bits are associated with the switch positions. Setting each switch position to ON or OFF sets the corresponding address bit to a logic 0 or a logic 1, respectively. The binary value corresponding to the switch position is listed below the switch illustration. The binary value is then converted to

its hexadecimal equivalent. This hexadecimal number is the base address of the SCB-1020.

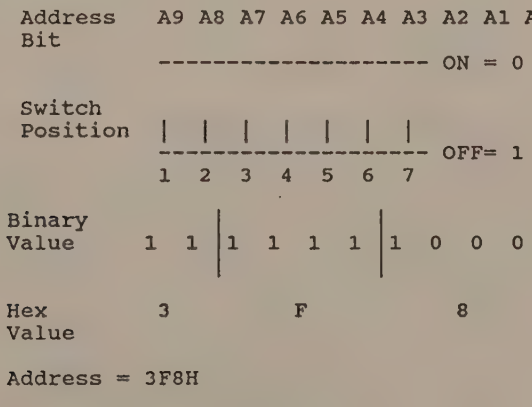
8254 AND μPD7201A INTERNAL REGISTERS

A2	A1	A0	REGISTER	
0	0	0	TIMER/COUNTER 0	8254
0	0	1	TIMER/COUNTER 1	8254
0	1	0	TIMER/COUNTER 2	8254
0	1	1	MODE CONTROL	8254
1	0	0	CHANNEL A DATA	μPD7201A
1	0	1	CHANNEL B DATA	μPD7201A
1	1	0	CHANNEL A CONTROL	PD7201A
1	1	1	CHANNEL B CONTROL	PD7201A

FIGURE 3

The SCB-1020 performs data transfers by using channel A of the μPD7201A.

Example 1:


Address Bit
A9 A8 A7 A6 A5 A4 A3 A2 A1 A0
ON = 0
Switch Position
OFF= 1
1 2 3 4 5 6 7
Binary Value
1 1 1 1 1 1 1 0 0 0
Hex Value
3 F 8
Address = 3F8H

Example 2:

```
Address   A9 A8 A7 A6 A5 A4 A3 A2 A1 A0
Bit
                                    ON = 0
Switch       |     |     |
Position  |     |     |     |
                                    OFF= 1
          1  2  3  4  5  6  7

Binary
Value     1  0 |1  0  1  0 |1  0  0  0

Hex
Value      2         A           8

Address = 2A8H
```

Example 3:

```
Address   A9 A8 A7 A6 A5 A4 A3 A2 A1 A0
Bit
                                    ON = 0
Switch    |  |              |
Position        |  |  |  |
                                    OFF= 1
          1  2  3  4  5  6  7

Binary
Value     0  0 |1  1  1  1 |0  0  0  0

Hex
Value      0         F           0

Address = 0F0H
```

(2) FACTORY ADDRESS SETTINGS

The SCB-1020 is shipped with an address of 210H. Switches 1 and 6 are in the off position while switches 2-5 and switch 7 are in the on position for this base address.

It is worth mentioning at this time the addresses IBM assigns to COM1 - COM4.

COM1 @ 3F8H
COM2 @ 2F8H
COM3 @ NOT DECLARED
COM4 @ NOT DECLARED

As the SCB-1020 utilizes the NEC µPD7201A, it cannot be configured as a standard asynchronous serial "COM" port. Register level programming of the µPD7201A is required for both asynchronous and synchronous operation.

B. CONFIGURATIONS

(1) INDIVIDUAL INTERRUPT SELECT-J1

J1 is a twelve pin jumper block which allows access to interrupts 2-7 off the computers bus. The interrupt vectors for the IBM PC/XT/AT are as follows:

IRQ2: 28-2BH
IRQ3: 2C-2FH
IRQ4: 30-33H
IRQ5: 34-37H
IRQ6: 38-3BH
IRQ7: 3C-3FH

IRQ3 and IRQ4 are the standard IBM BIOS COM2 and IBM BIOS COM1 interrupts respectively.

(2) DMA SELECT-J2

DMA channels 1 and 3 are available via J2. DMA channel 0 is used for memory refresh purposes and channel 2 is used for direct memory access by the diskette controller. To avoid any conflict with these operations, channel 0 and channel 2 are not available for user access. The SCB-1020 supports several DMA configurations to allow the user to easily implement the SCB-1020 with already existing DMA system configurations. The direction of data flow either from memory to the SCB-1020 or from the SCB-1020 to memory is determined by setting or clearing the data direction bit within the DMA controller chip (8237A). The SCB-1020 may be configured in the following modes:

a) request DMA transfer for transmission purposes on channel 1 by connecting pins 1&9, 3&11 of J2.

b) request DMA transfer for transmission purposes on channel 3 by connecting pins 2&10, 8&16 of J2.

c) request DMA transfer for reception purposes on channel 1 by connecting pins 1&9, 7&15 of J2.

d) request DMA transfer for reception purposes on channel 3 by connecting pins 6&14, 8&16 of J2.

e) request DMA transfer for both transmission and reception purposes on channel 1 by connecting pins 1&9, 5&13 of J2.

f) request DMA transfer for both transmission and reception purposes on channel 3 by connecting pins 4&12, 8&16 on J2.

(3) INTERRUPT MODE SELECT-J3

The six pin jumper, J3, makes available to the μPD7201A one of two possible interrupt schemes.

The first scheme is selected by connecting pins 1&4. This connection provides the μPD7201A access to interrupts 2-7 off the computers bus. Jumper block J1 then selects the desired interrupt from IRQ2-IRQ7.

The second scheme allows the user to share an interrupt line with one or more Qua Tech peripherals with equivalent interrupt sharing capabilities. This is achieved by connecting pins 2&5, and 3&6 of J3 on the SCB-1020 and only pin 2&5 of the jumper on another Qua Tech peripheral(s). This procedure may be reversed by connecting pins 2&5 of the SCB-1020 and pins 2&5, and 3&6 of the jumper on another Qua Tech peripheral. Please refer to all other Qua Tech's operation manuals that obtain these interrupt sharing capabalities to confirm proper jumper settings. FIGURE 4 illustrates the hardware to allow this interrupt sharing scheme. A software polling routine can then be implemented to determine which board generated the pending interrupt.

HARDWARE FOR INTERRUPT SHARING CAPABILITIES

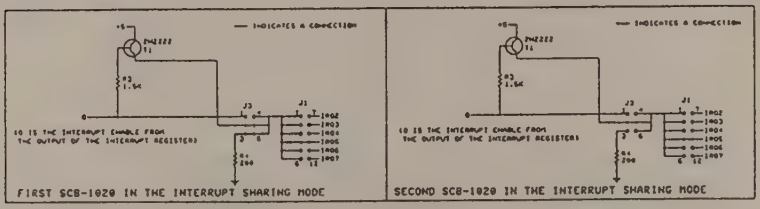


FIGURE 4

(4) EIA-232 DTE/DCE SELECT- J4

When using the SCB-1020 it is possible to configure it as either Data Terminal Equipment (DTE) or Data Communications Equipment (DCE). For DTE configuration connect pins 2 & 9, 3 & 10, 4 & 11, 5 &12, 6 & 13, and 7 & 14. For DCE operation connect pins 2 & 3, 4 & 5, 6 & 7, 9 & 10, 11 & 12, and 13 & 14.

Pin one provides a digital ground while pin 8 supplies a chassis ground. In the case where the computer chassis is not grounded, pins 1 & 8 can be connected to provide a chassis ground. FIGURE 5 and

FIGURE 6 illustrate DTE and DCE signal directions respectively. For a tabulated listing of DTE/DCE signal directions refer to FIGURE 7.

DTE SIGNAL DIRECTION

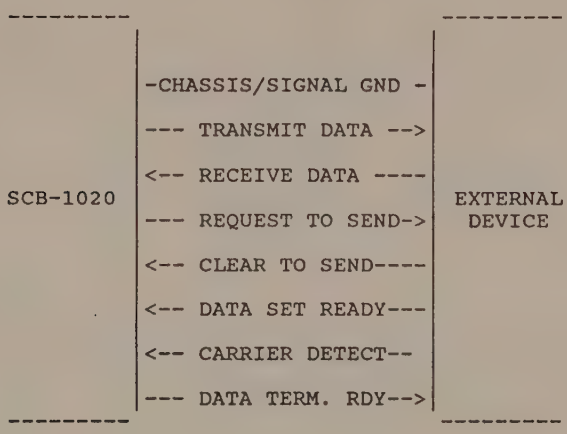


FIGURE 5

DCE SIGNAL DIRECTION

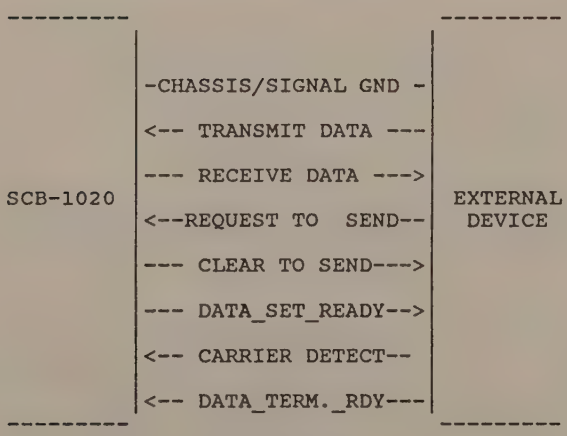


FIGURE 6

EIA-232 SIGNALS AND DIRECTIONS ON D-25

D-25 PIN #	EIA-232 SIGNAL NAME	DTE DIRECTION	DCE DIRECTION
1	CHASSIS GND	-	-
2	TXD	OUT	IN
3	RXD	IN	OUT
4	RTS	OUT	IN
5	CTS	IN	OUT
6	DSR	IN	OUT
7	SIGNAL GND	-	-
8	CD	IN	IN
20	DTR	OUT	IN

FIGURE 7

(5) INTERRUPT SOURCE AND RTS/CTS CONTROL SELECT-J5

The eight pin jumper block J5 provides two methods for operating in an interrupt driven mode and also two methods to apply the RTS/CTS signals.

Connecting pins 1&2 of J5 allows the μPD7201A to generate an interrupt which presets an on board interrupt register. The output of this register generates the desired interrupt (IRQ2 -IRQ7) as configured by J1 to the interrupt controller chip (8259A). The μPD7201A may generate an interrupt for any of the following cases:

a) interrupt on first received character.
b) interrupt on all received characters.
c) interrupt on transmit buffer empty.
d) interrupt on parity error.
e) interrupt on receiver overrun error.
f) interrupt on CRC framing error.
g) interrupt on end of SDLC frame.

If the μPD7201A is enabled for DMA operation (enable bit within the μPD7201A, refer to APPENDIX A for proper settings) it is still possible to generate an interrupt for any of the above mentioned cases.

An interrupt may be generated to the host system upon a DMA terminal count by connecting pins 5&6 of J5. A DMA terminal count (T/C) signal is provide to the SCB-1020 from the computer's bus and goes active upon the completion of a DMA operation. The terminal count signal presets the interrupt register whose output generates the desired interrupt (IRQ2-IRQ7) as determined by J1. The interrupt routine could then notify the operator that the DMA transfer is complete. Operation in a background mode is therefore possible to further enhance the SCB-1020's

capabilities.

It is the user's responsibility to reset the interrupt register by executing a read statement to the μPD7201A within the interrupt service routine. If this procedure is not followed then upon return from the interrupt service routine, an interrupt will still be pending from the interrupt register and the program will vector into the interrupt service routine again. An example assembly language program for an IRQ5 service routine is included in the programming section of the manual to offer the reader further insight into the above discussed subject matter.

There are two applications concerning the RTS/CTS signals. The first feature allows the handshake signals to either be looped back to one another or be sent out to the D-25 connector. By connecting pins 3&7 of J5, the RTS signal (pin 38 of the μPD7201A) is connected directly to the CTS signal (pin 39 of the μPD7201A).

The RTS and CTS signals also appear at the D-25 connector. This allows the user to send an active RTS or to receive an active CTS from an external device or to loop back the two handshake indicators from the connector. To connect these two signals to the connector attach pins 3 & 4, and 7 & 8 of J5.

(6) TRANSMIT CLOCK SOURCE SELECT-J6

The SCB-1020 may receive its synchronizing transmit clock from one of two sources.

The first source is provided by counter0/timer0 of the 8254 programmable counter/timer. The 8254 allows the user to divide any input clock signal by a value which is programmed into one of its counter/timers. The range of divisor values is from 0 to 65,535. For example, the 8254 receives a clock input of 5MHz at the CLK0 input (pin 9) and if a divisor of 500 is programmed into counter0/timer0 of the 8254, then an output clock of 50KHz is produced for use as the synchronizing transmitting clock. The μPD7201A may obtain this clock as its synchronizing transmitting clock by connecting pins 2 & 3 of J6.

The second source of the synchronizing transmitting clock may be provided by an external source. The μPD7201A may use this external clock by connecting pins 1&2 of J6.

The maximum frequency for the synchronizing transmitting clock is determined by the distance data is transferred, the environment, and type of cabling used. For distances up to approximately ten feet, the SCB-1020 has functioned properly at a clock rate of 250KHz in the asynchronous mode of operation. For

reliable EIA communication procedures, refer to specifications in FIGURE 2 of page 4.

(7) RECEIVE CLOCK SOURCE SELECT-J7

The SCB-1020 may obtain its receiving synchronization clock from one of two sources.

The first source is provided by timer1 of the 8254 programmable counter/timer. Again the 8254 allows the user to divide any input clock signal by a value form 0 to 65,535. The output of counter1/timer1 is then available as the synchronizing transmit clock. This clock may be input to the μPD7201A by connecting pins 2&3 of J7.

The second source of the synchronizing transmit clock may be provided by an external source. This external clock may be made available to the μPD7201A by connecting pins 1&2 of J7.

Again it must be stated that any synchronizing receive clock over the EIA specifications will not guarantee reliable communication.

(8) -SYNC/IPS SELECT-J8

The three pin jumper block, J6, allows the user to select one of three options. When operating in an asynchronous mode, pins 1&2 of J8 may be connected to provide an additional handshake signal to the μPD7201A. A typical application may be to use this signal as a Data Set Ready (DSR) input. When operating in an external synchronization mode, pins 1&2 of J8 may be connected to allow an external signal to be input to the μPD7201A which notifies it that synchronization has been achieved. In internal synchronization modes (monosync,bisync, and SDLC) pins 2&3 of J8 may be connected to provide an output signal which notifies an external device that a SYNC character match has been made.

A tabulated representation of all jumper configurations is presented in configurations are given in part (9) titled as JUMPER SUMMARY TABLE.

(9) JUMPER SUMMARY TABLE

J1-INDIVIDUAL INTERRUPT SELECT

FUNCTION	CONNECT
IRQ2 SELECT	1&7
IRQ3 SELECT	2&8
IRQ4 SELECT	3&9
IRQ5 SELECT	4&10
IRQ6 SELECT	5&11
IRQ7 SELECT	6&12

J2-DMA SELECT

FUNCTION	CONNECT
DMA TRANSFER FOR TRANSMISSION-CH1	1&9 3&11
DMA TRANSFER FOR TRANSMISSION-CH3	2&10 8&16
DMA TRANSFER FOR RECEPTION-CH1	1&9 7&15
DMA TRANSFER FOR RECEPTION-CH3	6&14 8&16
DMA TRANSFER FOR TRANSMISSION AND RECEPTION-CH1	1&9 5&13
DMA TRANSFER FOR TRANSMISSION AND RECEPTION-CH3	4&12 8&16

J3-INTERRUPT MODE SELECT

FUNCTION	CONNECT
INDIVIDUAL INTERRUPT WITHOUT POLLING	1&4
INDIVIDUAL INTERRUPT WITH POLLING	2&5* 3&6

* REFER TO SECTION III.B.(3) FOR PROPER INTERRUPT SHARING POSSIBILITIES

FIGURE 8

(9) JUMPER SUMMARY TABLE con't

J4-EIA-232 DTE/DCE CONFIGURATIONS

FUNCTION	CONNECT
CHASSIS TO DIGITAL GND	1&8
DTE	2&9
	3&10
	4&11
	5&12
	6&13
	7&14
DCE	2&3
	4&5
	6&7
	9&10
	11&12
	13&14

J5-INTERRUPT SOURCE & RTS/CTS CONTROL

FUNCTION	CONNECT
FROM μPD7201A	1&2
FROM DMA TERMINAL COUNT	5&6
RTS/CTS CONNECTIONS TO D-25	3&4
	7&8
RTS/CTS LOOPBACK	3&7

J6-TRANSMIT CLOCK SOURCE

FUNCTION	CONNECT
EXTERNAL TRANSMIT CLOCK	1&2
INTERNAL TRANSMIT CLOCK	2&3

J7-RECEIVE CLOCK SOURCE

FUNCTION	CONNECT
EXTERNAL RECEIVE CLOCK	1&2
INTERNAL RECEIVE CLOCK	2&3

FIGURE 8 con't

(9) JUMPER SUMMARY TABLE

J8--SYNC/IPS

FUNCTION	CONNECT
GENERAL PURPOSE INPUT	1&2
EXTERNAL SYNC MODE-SYNDET	1&2
INTERNAL SYNC MODE-SYNDET	2&3

FIGURE 8 con't

(10) FACTORY JUMPER CONFIGURATIONS

The SCB-1020 is shipped with the following jumper configurations:

J1	J2	J3	J4	J5	J6	J7	J8
3&9	1&9 3&11 6&14 8&16	1&4	1&8 2&9 3&10 4&11 5&12 6&13 7&14	1&2 3&4 7&8	2&3	2&3	2&3

FIGURE 9

J1 : IRQ4 is selected as individual interrupt.

J2 : DMA transfer for transmission on channel 1. DMA transfer for reception on channel 3

J3 : Individual interrupt without polling.

J4 : DTE,Chassis Ground is connected to Digital Ground.

J5 : μPD7201A is interrupt source. RTS/CTS are available at D-25.

J6 : Transmit Clock Source is internal.

J7 : Receive Clock Source is internal.

J8 : Internal Sync Mode-Syndet output. General purpose input for asynchronous and an input to indicate synchronization in external sync mode.

C. EXTERNAL CONNECTIONS

Connections to the outside world are made available via a D-25 connector. This type of connector has been proven effective in providing excellent shielding against noise generated from environmental conditions. A pin-out of the D-25 connector is illustrated in FIGURE 10. A detailed description of each signal is provided for convenient reference where the signal name and the pin number in parenthesis are given.

D-25 PIN-OUT

NC	13	25	GPCO
NC	12	24	TXCO
NC	11	23	NC
NC	10	22	NC
NC	9	21	NC
-CD	8	20	DTR
DGND	7	19	-SYND
IPS	6	18	NC
CTS	5	17	RXCI
RTS	4	16	NC
RXD	3	15	TXCI
TXD	2	14	RXCO
CGND	1		

FIGURE 10

CGND (1)-

THIS GROUND SIGNAL IS CONNECTED TO THE COMPUTER'S CHASSIS AND IF PINS 1&8 OF J8 ARE CONNECTED,IT IS ALSO CONNECTED TO DIGITAL GROUND. CHASSIS AND DIGITAL GROUND ARE AVAILABLE FOR AN EXTERNAL DEVICE.

TXD (2)-

TRANSMIT SERIAL DATA

RXD (3)-

RECEIVE SERIAL DATA

RTS (4)-

REQUEST TO SEND. INFORMS THE MODEM OR AN EXTERNAL DEVICE THAT THE PD2701A IS READY TO SEND DATA.

CTS (5)-

CLEAR TO SEND. INFORMS THE μPD7201A THAT THE MODEM OR AN EXTERNAL DEVICE IS READY TO RECEIVE DATA.

IPS (6)-

INPUT SIGNAL. ACTIVE "LOW" INPUT TO μPD7201A TO INDICATE SYNCHRONIZATION FOR EXTERNAL SYNC MODE OR A GENERAL PURPOSE INPUT FOR ASYNCHRONOUS MODE SUCH AS THE DSR SIGNAL.

DGND (7)-

DIGITAL GROUND.

NC (8)
(9)
(10)
(11)
(12)
(13)-

NO CONNECTION

RXCO (14)-

RECEIVE CLOCK OUT. PROVIDED BY COUNTER1/TIMER1 OF THE 8254. DATA IS RECEIVED ON THE RISING EDGE OF THE INTERNAL RECEIVE CLOCK.

TXCI (15)-

TRANSMIT CLOCK IN. SUPPLIED BY AN EXTERNAL DEVICE. DATA IS TRANSMITTED ON THE FALLING EDGE OF THE EXTERNAL TRANSMIT CLOCK.

NC (16)-

NO CONNECTION

RXCI (17)-

RECEIVE CLOCK IN. SUPPLIED BY AN EXTERNAL DEVICE. DATA IS RECEIVED ON THE RISING EDGE OF THE EXTERNAL RECEIVE CLOCK.

NC (18)-

NO CONNECTION

SYNDET (19)-

USING BYTE SYNCHRONOUS MODE FOR INTERNAL SYNC DETECT OR USING SDLC MODE INDICATING FLAG DETECTION. THIS SIGNAL IS USED FOR OUTPUT ONLY.

DTR (20)-

DATA TERMINAL READY. INFORMS MODEM OR DATA SET THAT μPD7201A IS READY TO RECEIVE INFORMATION.

NC (22)
(23)-

NO CONNECTION

TXCO (24)

TRANSMIT CLOCK OUT. PROVIDED BY COUNTER\TIMER0 OF THE 8254. DATA IS TRANSMITTED ON THE FALLING EDGE OF THE INTERNAL TRANSMIT CLOCK.

GPCO (25)

GENERAL PURPOSE CLOCK OUT. FURNISHED BY COUNTER\TIMER2 OF THE 8254. THIS CLOCK PULSE MAY BE USED TO START OR STOP AN EXTERNAL DEVICE.

IV. PROGRAMMING THE SCB-1020

A. OPERATIONAL DESCRIPTION

This section presents general information on the programmable hardware residing on the SCB-1020. The 8254 programmable counter/timer and the μPD7201A multiprotocol serial controller will be the topics of interest. For more detailed information regarding these two integrated circuits refer to the sections entitled "DETAILED INFORMATION ON THE 8254" and "DETAILED INFORMATION ON THE μPD7201A". This section is concluded with several example programs which offers for a majority of discussed topics.

(1) 8254 PROGRAMMABLE COUNTER/TIMER

The 8254 contains three individual sixteen bit counter/timers. Each counter/timer is capable of handling a maximum clock input of 10MHz. All three counters/timers are configured through a common mode control register.

The SCB-1020 uses the 8254 as a means of generating both the synchronizing transmit and receive clocks. In addition the on board 8254 supplies the user with the general purpose clock. A 5 MHz signal is provided to each counter/ timer which may be divided down to the desired output frequency. by programming the proper divisor value in each counter/ timer. For example, it is desired to have a 1MHz signal as the output from counter0/timer0,a 500KHz signal as the output from counter1/timer1, and finally a 100Hz signal as the output from counter2/timer2. To determine the divisor value for each counter we apply the following formula:

divisor value=5MHz/desired output freq.

Therefore, for counter0/timer0 a divisor of (5MHz/1MHz=5) 5 should be used. For counter1/timer1, a divisor of (5MHz/500KHz=10) 10 would be applied and for counter2/timer2, a divisor of (5MHz/100Hz=50000) 50000 would produce desired results. Each divisor value can be loaded into the appropriate counter/timer by simply writing to the mode control register which determines the counter to be loaded or read and the mode of operation in which the counter/timer is to be used. After writing to the mode control register the divisor value may be written to the appropriate counter/timer. Each counter/timer may be operated in one of six possible modes. The available modes
of operation are as follows:

1) Mode 0: Interrupt on terminal count
2) Mode 1: Programmable one-shot
3) Mode 2: Rate generator
4) Mode 3: Square wave generator

5) Mode 4: Software triggerable start
6) Mode 5: Hardware triggerable start

(2) μPD7201A MULTIPROTOCOL SERIAL CONTROLLER

The μPD7201A is a two channel multiple communication controller capable of supporting asynchronous, byte synchronous, and bit synchronous communication. The SCB-1020 utilizes channel A of the μPD7201A to transfer data. The μPD7201A functions as a parallel to serial and serial to parallel converter/controller. The SCB-1020 is specified of a maximum transfer rate of 20 Kbaud. Data rates of up to 250 Kbps have been achieved for short distances.

The μPD7201A contains one channel A transmit register, one channel B transmit register, one channel A receive register, one channel B receive register, seven channel A control registers, seven channel B control registers, two channel A status register, and five channel B status registers. The addressing scheme for accessing the above register locations can be simplified in the following way: Address bit A0 selects the desired channel: A0=0 Channel A, A0=1 Channel B. Address bit A1 selects either the data registers or the control /status registers: A1=0 Data registers, A1=1 Control/Status registers. This addressing scheme is summarized in FIGURE 9. Data to be transmitted or received is accessed through the data registers. For example, if data is to be output to an external device it is written to the data register. If data is to be received from an external device then it is read from the data register. The data register is straight forward, however the control and status registers are a bit more involved. It was stated above that there are seven control registers for each channel, two status registers for channel A, and five status registers for channel B. One might now ask how can so many locations be addressed by only two address bits A0 and A1. This is a fair question and may best be answered as follows: when A1=1 the control/status registers are accessed. Write operations are for control and read operations are for status. The three least significant bits of the control/status register are used to address the particular control/status register one is interested in. Consider the following example: It is desired to write 255 to control register 5 of channel A. A0=0 for Channel A and A1=1 for control/status registers. Since we are performing a write operation we will access the control registers. We write the number 5 to the control register. The three least significant bits of the control register now contain the data "5". This specifies which register number is to be accessed on the next control register write or status register read. We therefore write 255 to the control register and it is automatically written to control register 5.

The three least significant bits of the control/status register therefore act as a register pointer which specifies which register the data is to written to or read from. The following table lists the various control and status registers.

μPD7201A CONTROL AND STATUS REGISTERS

CONTROL REGISTER (CHANNEL A & B)	FUNCTION
0	SYSTEM COMMANDS AND REGISTER POINTER
1	INTERRUPT CONTROL
2	INTERFACE CONTROL
3	RECEIVER CONTROL
4	MODE CONTROL
5	TRANSMITTER CONTROL
6	SYNC BYTE 1
7	SYNC BYTE 2

STATUS REGISTER (CHANNEL A)	FUNCTION
0	BUFFER STATUS
1	LINE CONTROL STATUS
3	TXD COUNT LOW BYTE
4	TXD COUNT HIGH BYTE

STATUS REGISTER (CHANNEL B)	FUNCTION
0	BUFFER STATUS
1	LINK CONTROL STATUS
2	INTERRUPT VECTOR
3	TXD COUNT LOW BYTE
4	TXD COUNT HIGH BYTE

FIGURE 11

B. PROGRAMMING EXAMPLES

(1) Example 1

```
 10 '************************************************
 20 '
 30 'THIS PROGRAM CONFIGURES BOTH COUNTER0 AND
 35 'COUNTER1 TO OUTPUT A 1MHz SQUARE WAVE TO BE
 40 'USED AS THE INTERNAL SYCHRONIZING CLOCKS FOR
 50 'THE TRANSMITTER AND RECEIVER ON THE SCB-1020.
 60 'COUNTER2 IS USED AS THE GENERAL PURPOSE CLOCK
 65 'AND OUTPUTS A 500KHz SQUARE WAVE.THE BASE ADDRESS
 70 'OF THE SCB1040 IS 300H.
 75 'SELECT SQUARE WAVE GENERATOR AS THE MODE FOR THE
 80 '8254.
 90 '
100 '************************************************
110 '
120 OUT &H303,&H36    'configure counter 0
130 OUT &H300,5       'set lsb of counter0  with divisor
                      'of 5
140 OUT &H300,0       'set msb of counter0 with divisor
                      'of 0
150 OUT &H303,&H76    'configure counter 1
160 OUT &H301,5       'set lsb of counter1 with divisor
                      'of 5
170 OUT &H301,0       'set msb of counter1 with divisor
                      'of 0
180 OUT &H303,&HB6    'configure counter2
190 OUT &H302,10      'set lsb of counter2 with divisor
                      'of 10
200 OUT &H302,0       'set msb of counter2 with divisor
                      'of 0
210 END
```

(2) Example 2

```
1 '************************************************
2 '
3 'THIS PROGRAM PERFORMS A SIMPLE TRANSMISSION TEST OF
4 'THE SCB-1020'S TRANSMIT CIRCUITRY. THE BASE ADDRESS
5 'OF THE BOARD IS 3F0H. THE SYNC CLOCKS ARE RUNNING AT
6 '1MHz.
7 '
8 '************************************************
10 OUT &H3F3,&H36     'configure counter 0
20 OUT &H3F0,5        'set lsb of counter0 with divisor
                      'of 5
30 OUT &H3F0,0        'set msb of counter0 with divisor
                      'of 0
40 OUT &H3F3,&H76     'configure counter 1
50 OUT &H3F1,5        'set lsb of counter1 with divisor
                      'of 5
60 OUT &H3F1,0        'set msb of counter1 with divisor
                      'of 0
70 CLS
80 OUT &H3F6,4        'access wr4
90 OUT &H3F6,&H48     '16X data rate=clock rate, no parity,
```

```
                        '1.5 stop bits
100 OUT &H3F6,3         'access to wr3
110 OUT &H3F6,&HC1      'select 8 data bits at wr3 for rx
                        'buffer,enable rx
120 OUT &H3F6,5         'access to wr5
130 OUT &H3F6,&HEA      'select 8 data bits at wr5 for tx
                        'buffer,enable tx
140 OUT &H3F6,1         'access wr1
150 OUT &H3F6,2         'no interrupts or dma
160 OUT &H3F6,2         'access wr2
170 OUT &H3F6,&H0       'no interrupts or dma
180 FOR I=0 TO 255
190 OUT &H3F4,I         'send loop variable
200 C%=INP(&H3F6)       'get status
210 D%=C% AND 1         'mask data ready bit
220 IF D%=1 GOTO 240 'if dr is set then send another
                        'character
230 GOTO 200            'fetch status again
240 DATA1%=INP(&H3F4)'receive character
250 IF DATA1%=I GOTO 270 'char.should be that just sent
260 GOTO 200            'if false check for character
270 NEXT I              'if true send next character
280 LOCATE 10,30:PRINT "********************"
290 LOCATE 11,30:PRINT "END OF TRANSMISSION"
300 LOCATE 12,30:PRINT "********************"
310 END
```

(3) Example 3

```
10
'*****************************************************
20 '
30 'THIS PROGRAM PERFORMS A DMA TRANSFER USING
35 'CHANNEL3. DATA IS PLACED IN MEMORY STARTING
40 'AT 3000:0H. THIS DATA IS ACQUIRED FROM THE
45 'SCB-1020 VIA DMA. THE BASE ADDRESS OF THE
50 'SCB-1020 IS 300H. ABBREVIATE FLIP FLOP BY F/F.
60 '*****************************************************
70 '
80  DIM A(300)      'ARRAY FOR MEMORY DATA
90 '
145 '*************************************
146 '       DEFINE DMA ADDRESSES
147 '*************************************
150 BASEADD=&H6     'CH3 BASE ADDRESS
160 PAGE1=&H82      'CH3 PAGE REGISTER
170 BASEWC=&H7      'CH3 BASE WORD COUNT
180 RSTAT=&H8       'CH3 STATUS REGISTER
190 WCOMMAND=&H8    'CH3 COMMAND REGISTER
200 WREQUEST=&H9    'CH3 REQUEST REGISTER
210 WSINMASK=&HA    'CH3 MASK REGISTER BIT
220 WMODE=&HB       'CH3 MODE REGISTER
230 CLEARFF=&HC     'CLEAR BYTE PTR F/F
240 RTEMP=&HD       'READ TEMP.REGISTER
250 WMASTERC=&HD    'WRITE MASTER CLEAR
255 WALLMASK=&HF    'ALL MASK REG. BITS
260 '
```

```
'************************************
'   INITIALIZATION  DMA CONTROLLER
'************************************
'
OUT WSINMASK,7 'MASK DREQ3
OUT CLEARFF,0  'PUT F/F KNOW STATE
OUT BASEADD,&H0'LOC. LSB TO TRANSFER
OUT BASEADD,&H0'LOC. MSB TO TRANSFER
OUT PAGE1,3    'SET PAGE REGISTER
OUT WMODE,&H47 'SINGLE,CH3,ADDRESS, INC.,WRITE TRANSFER
OUT WREQUEST,0 'DISABLE INTERNAL SOFTWARE REQUEST
OUT CLEARFF,0
OUT BASEWC,&HFF'TRANSFER 255 BYTES
OUT BASEWC,&H0
'
'************************************
'          CONFIGURE µPD7201A
'************************************
'
OUT &H303,&H36  'configure counter 0
OUT &H300,5      'set lsb of counter0 with divisor
OUT &H300,0      'set msb of counter0 with divisor
                 'of 0
OUT &H303,&H76  'configure counter 1
OUT &H301,5      'set lsb of  counter1 with divisor
                 'of 5
OUT &H301,0      'set msb of counter1 with divisor
                 'divisor of 0
OUT &H306,&H18   'reset channel A
OUT &H307,&H18   'reset channel B
FOR I=0 TO 50:NEXT I 'time delay
OUT &H306,2      'access wr2
OUT &H306,3      'select dma mode
OUT &H306,4      'access wr4
OUT &H306,&H48   '16X data rate=clock, rate, no
                 'parity, 1.5 stop bits
OUT &H306,3      'access to wr3
OUT &H306,&HC1   'select 8 data bits at wr3 for
                 'rx buffer, enable rx
OUT &H306,5      'access to wr5
OUT &H306,&H68   'select 8 data bits enable tx
OUT &H306,1      'access wr1
OUT &H306,8      'enable dma
OUT WSINMASK,3   'unmask DREQ3
FOR I= 0 TO 255
OUT &H304,I
NEXT I
CLS
PRINT " DATA INSTALLED IN MEMORY"
PRINT " "
DEF SEG=&H3000
FOR I=0 TO 255
PRINT PEEK(I);
NEXT I
END
```

(4) Example 4

```
;*****************************************************
;THIS PROGRAM IS AN INTERRUPT 5 SERVICE ;ROUTINE TO
;RESET THE INTERRUPT GENERATED DMA TERMINAL COUNT
;*****************************************************
;
STACK SEGMENT PARA STACK 'STACK'
DB 64 DUP(?)
STACK ENDS
;
RESIDENT SEGMENT PARA PUBLIC 'CODE'
ASSUME CS:RESIDENT,DS:RESIDENT
JMP A4
;
;*****************************************************
;      INTERRUPT SETUP PROCEDURE
;*****************************************************
;
INTSETUP PROC FAR
ASSUME CS:RESIDENT,DS:RESIDENT
XOR AX,AX      ;CLEAR AX
PUSH ES        ;SAVE ES
MOV ES,AX      ;ES=0
MOV DI,34H     ;DI=IRQ5 ADDRESS
MOV AX,OFFSET SERVICE
MOV ES:[DI],AX;STORE 1ST BYTE OF SERVICE
MOV AX,CS      ;AX=CURRENT CODE SEG
MOV ES:[DI]+2,AX;STORE 2ND BYTE = CS
IN AL,21H      ;FETCH INTERRUPT MASK REG.
AND AL,0DFH    ;ENABLE IRQ5
OUT 21H,AL     ;OUTPUT MASK BYTE TO IMR
POP ES         ;RESTORE EXTRA SEGMENT
RET
INTSETUP ENDP
;


;*************************************
; IRQ5 SERVICE ROUTINE TO CLEAR ON BOARD
; FLIP FLOP SET BY DMA TERMINAL COUNT.
;*************************************
;
SERVICE PROC NEAR
PUSH AX        ;SAVE SYSTEM REGISTERS
PUSH SI
PUSH ES
PUSH DX
PUSH DS
PUSH DI
PUSH CX
PUSH BX
PUSH BP
PUSHF
IN AL,21H      ;FETCH INTERRUPT MASK BYTE
OR AL,20H      ;DISABLE IRQ5
OUT 21H,AL     ;OUTPUT COMMAND BYTE
MOV DX,306H    ;DX=ADDRESS OF SCB-1020
```

```
IN AL,DX      ;CLEAR FLIP FLOP BY READ
MOV AL,20H    ;RESET IRQ CONTROLLER CHIP
OUT 20H,AL
IN AL,21H     ;FETCH IMR BYTE
AND AL,0DFH   ;ENABLE IRQ5
OUT 21H,AL
POPF          ;RESTORE SYSTEM REGISTERS
POP BP
POP BX
POP CX
POP DI
POP DS
POP DX
POP ES
POP SI
POP AX
RETI
SERVICE ENDP
;
;*******************************************************
; SECTION OF CODE TO PROTECT IRQ ROUTINE
;*******************************************************
;
A4:
ADDRESS EQU 180H  ;ESTABLISH VECTOR ADD.
PUSH DS           ;SAVE RETURN SEGMENT
XOR AX,AX         ;CLEAR AX
PUSH AX           ;SAVE AX
MOV BX,1          ;BX=VECTOR LOCATION
MOV BYTE PTR[BX],27H ;27H FOR RESIDENCY
MOV DS,AX         ;DS=0
MOV AX,CS         ;AX=CODE SEGMENT
MOV DS:ADDRESS,AX ;STORE CODE SEGMENT
MOV DX,OFFSET A4+100H ;JUMP OVER CODE
DUMMY PROC FAR    ;EXECUTE RETURN
RET
DUMMY ENDP
RESIDENT ENDS
END
```

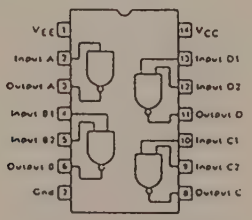
V. EIA STANDARD SUPPORTED BY THE SCB-1020

A. EIA-232 STANDARD

The SCB-1020 supports the EIA-232 standard by using appropriate driver/receiver sets. One of the driver/receiver sets that the SCB-1020 uses is Motorola's MC1489 quad line receiver with input range of 30 Vdc and a Motorola MC1488 quad line driver with an output limited to 10 mA (typical). Both the receiver and driver integrated circuits meet the specifications for the Electronic Industries Associations Standard No. EIA-232C. These devices are designed to interface data terminal equipment with data communications equipment. A pin out of Motorola's driver and receiver are shown in FIGURE 12.

The distance to drive or receive signals may vary depending on various factors such as the capacitance on the cable, the type of the environment the SCB-1020 is in, and the baud rate. For instance, if the cable that connects the SCB-1020 and an external device has a capacitance value of 2500 pF, then the two devices could communicate at a distance of approximately 50 feet at a baud rate of approximately 20kbaud. The SCB-1020 is capable of transmitting and receiving these distances with the assistance of drivers and receivers that meet EIA specifications.

MC1488



MC1489

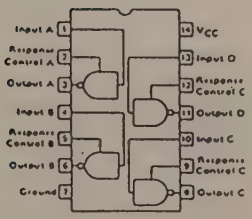


FIGURE 12

VI. DETAILED INFORMATION ON THE 8254

A pin-out of the 8254 is illustrated in FIGURE 13. The 8254 contains 3 sixteen bit counters/timers. Each counter/timer is controlled through a common mode control register. The mode control format is illustrated in FIGURE 14.

PINOUT OF 8254

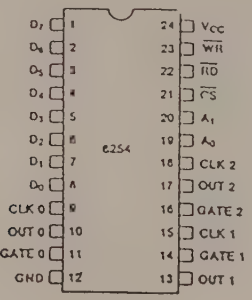


FIGURE 13

8254 CONTROL WORD FORMAT

SC1	SC0	RL1	RL0	M2	M1	M0	BCD

where:

SC1	SC0	FUNCTION
0	0	COUNTER0
0	1	COUNTER1
1	0	COUNTER2
1	1	ILLEGAL

M2	M1	M0	FUNCTION
0	0	0	MODE0
0	0	1	MODE1
X	1	0	MODE2
X	1	1	MODE3
1	0	0	MODE4
1	0	1	MODE5

RL1	RL0	FUNCTION
0	0	LATCH DESIRED COUNTER FOR OPERATION ON THE FLY
0	1	READ/LOAD LEAST SIG.BYTE
1	0	READ/LOAD MOST SIG.BYTE
1	1	READ/LOAD LEAST SIG.BYTE FIRST THEN MOST SIG.BYTE

BCD	FUNCTION
0	BINARY 16 BIT COUNTER
1	BCD COUNTER

FIGURE 14

Each counter/timer must be configured before an initial count value can be loaded into it. The counter to be configured is determined by SC0 and SC1. RL0 and RL1 determine the method that the counter/timer is to be loaded and read. It is possible to utilize each counter/timer as two eight bit counter/timers as opposed to one sixteen bit counter/timer. For this reason it is necessary to load the least significant byte and the most significant byte of each counter/timer separately. When using each counter/timer as sixteen bit, it is convenient to automatically load the least significant byte and then the most significant byte. This is done by setting both RL0 and RL1 to "1". M0, M1, M2 determine the mode of operation that each counter/timer is to be used in. The six possible modes of operation are as follows:

1) MODE0-Interrupt on terminal Count:

The output is initially low upon the configuring and loading of the counter/timer. The counter/timer decrements its count to zero. Upon reaching zero the output goes high and remains high until the counter/timer is reconfigured or reloaded.

2) MODE1-Programmable one-shot:

Upon the rising edge of the gate input signal the output goes low and the counter/timer starts to count. Upon terminal count the output goes high. The one-shot is retriggerable which means that the rising edge of any further gate input will force the output to a low state.

3) MODE2-Rate generator:

The output is initially low after the first period of the input clock. Upon terminal count the output goes high and then goes low on the next period of the input clock. The counter/timer is automatically reloaded with the initial count value and the process continues. This mode allows the counter/timer to function as a divide by N counter.

4) MODE3-Square wave generator:

The output is initially low after the first clock period of the input clock. The counter/timer is decremented by two upon each input clock signal. Upon terminal count the output changes state. The counter/timer is reloaded with the initial count value and the process repeats.

5) MODE4-Software triggered start:

The output will go high upon configuring of the counter/timer. Upon terminal count the output will go low for one input clock period and then go high again.

6) MODE5-Hardware triggered start:

The output will go high upon configuring of the counter/timer. Upon the rising edge of gate input the counter/timer will start counting. Upon terminal count the output will go low for one input clock period and then go high again.

The BCD bit allows each counter/timer to be operated as either a sixteen bit binary counter or as a four decade binary coded decimal counter.

The location of each counter/timer and the mode control register is given by the base address of the SCB-1020 plus the following information:

A2	A1	A0	LOCATION
0	0	0	COUNTER 0
0	0	1	COUNTER 1
0	1	0	COUNTER 2
0	1	1	MODE CONTROL REGISTER

Further information on the 8254 programmable counter/timer may be obtained by referring to an INTEL Microprocessor and Peripheral Data Handbook.

VII. DETAILED INFORMATION ON THE μPD7201A

The μPD7201A uses two address lines to access its internal registers. Address bit A0 (pin 25) is used to determine the channel and address bit A1 (pin 24) selects either the data register or the control/status registers. The locations of the above registers are given by the base address of the SCB-1020 and the following information:

A2	A1	A0	REGISTER NAME
1	0	0	CHANNEL A DATA REGISTER
1	0	1	CHANNEL B DATA REGISTER
1	1	0	CHANNEL A CONTROL/STATUS
1	1	1	CHANNEL B CONTROL/STATUS

The data registers are used to transmit and receive information. To transmit information, data is written to the data register and to receive information, data is read from the data register. The control/status register utilizes the three least significant bits to select one of the eight possible control registers or one of the two possible status registers (five possible for channel B). A functional description of each control and status register is explained in the following subsections.

A. CONTROL REGISTER 0

Control register 0 has three functions. It is used as the register pointer, CRC control, and for issuing various commands. These commands consist of channel reset, enable interrupt on next character, reset pending transmitter interrupt/DMA request, error reset and end of interrupt.

B. CONTROL REGISTER 1

Control register one sets specific interrupt conditions and also sets or clears transmit count mode enable.

C. CONTROL REGISTER 2A

Control register two sets the interfacing mode of operation and what interrupt vector is used. The interface mode of operations consist of non-DMA or DMA. Possible interrupt vectors include non-vectored, 8085 Master, 8085 Slave, 8086, or 8085/8259A Slave.

D. CONTROL REGISTER 2B

Control register two B sets the interrupt vector when operating in the vectored interrupt mode.

E. CONTROL REGISTER 3

Control register three controls the receiver configurations.

F. CONTROL REGISTER 4

Control register four configures the clock rate, protocol mode, and parity. The clock rate actually configures the data rate as clock rate ÷1, clock rate ÷ 16, clock rate ÷32, or clock rate ÷ 64. The protocol modes consist of monosync, bisync, SDLC, external synchronization, or asynchronous operations.

G. CONTROL REGISTER 5

Control register five controls the transmitter configurations.

H. CONTROL REGISTER 6

Control register six represents a sync character or secondary address if operation in the SDLC mode. The function of this sync character depends on which protocol mode is being used.

I. CONTROL REGISTER 7

Control register seven represents a sync character or the SDLC synchronous bit sequence for SDLC. The function of this sync character depends on if using the monosync or the bisync protocol.

J. STATUS REGISTER 0

Status register zero contains the status of the receiver, the transmitter, and external flags. These flags consist of the Break/Abort flag, Idle/CRC flag, CTS flag, the Sync Status flag, and the DCD flag.

K. STATUS REGISTER 1

Status register one indicates the status of the line control conditions. These conditions consist of end of SDLC Frame, CRC framing error, Overrun error, Parity error, and the SDLC residue code.

L. STATUS REGISTER 2B

Status register two B contains either the interrupt vector progammed in control register two B or (if a condition affects mode is enabled) the modified interrupt vector. This modified interrupt vector informs the user what caused the interrupt.

M. STATUS REGISTER 3

Status register three stores the least significant byte of the transmit word count register once it is programmed after enabling the transmit byte count mode of control register one.

N. STATUS REGISTER 4

Status register four stores the most significant byte of the transmit word count register once it is programmed after enabling the transmit byte count mode of control register one.

Please refer to APPENDIX A for a complete explanation on how to configure each control register and what each bit represents in the status registers.

VIII. ANSWERS TO COMMONLY ASKED QUESTIONS

Question:

Is it possible to use the SCB-1020 as an EIA-232 asynchronous communication board?

Answer:

Yes. It should be mentioned that most existing communication software out in the market has been written for the 8250 and 16450 line of UART's. It may therefore be necessary to write one's own application software.

Question:

Is it possible to operate the SCB-1020 in a DMA background mode?

Answer:

Yes. An interrupt may be generated upon the DMA terminal count. A properly written service routine may inform the operator that a DMA process is finished.

Question:

What is it meant by sharable interrupts?

Answer:

A sharable interrupt is an interrupt that is used by two or more devices. Proper hardware must exist on each device which shares the interrupt to prevent any subsequent damage. Consider the following: two devices are sharing interrupt 5 (IRQ5), the interrupt signal is held low when inactive and goes to a high state when an interrupt is generated. If proper hardware does not exist on both boards,when one generates an interrupt there is a short circuit between the two devices and a large current will flow causing possible damage. An interrupt polling procedure must be applied to determine the origin of the generated interrupt.

Question:

What is the difference between Bisync, HDLC and SDLC synchronous communication.

Answer:

Bisync (Binary Synchronous Communication) is a byte oriented, half duplex protocol which transmits messages consisting of strings of characters. Error checking is performed only on the transmission block. Transmission speed is much slower then that of bit oriented protocols. HDLC (High-Level Data Link Control) is a

full duplex, bit oriented protocol which transmits messages consisting of frames. Error checking is performed on the entire frame. High data transfer speeds may be obtained through bit oriented protocols.SDLC (Synchronous Data Link Control) is a subset of the HDLC protocol.

Question:

What is the maximum data transfer rate the SCB-1020 will support?

Answer:

A 250 Kbps data transfer rate is possible by driving the μPD7201A at a clock rate of 250 KHz. However, the SCB-1020's physical layer is based on the EIA-232 Serial Bus Specifications as shown in FIGURE 2 on page 4. However, higer data rates have been achieved for various short distances. Therefore, it is under the user's discretion to choose the proper data rate depending on his/her application.

Question:

After carefully reading the SCB-1020 instruction manual, there are several questions which are still unanswered. What course of action should be taken?

Answer:

As always, Qua Tech, Inc. is dedicated to analyzing and solving all customer questions and problems. For answers to questions concerning the SCB-1020 contact Qua Tech's technical support staff at (216) 434-4479.

IX. APPENDIX A

A. NEC'S DATA SPECIFICATIONS ON THE μPD7201

NEC'S DATA SPECIFICATIONS IS REPRODUCED WITH PERMISSION OF NEC ELECTRONICS.

NEC
NEC Electronics Inc.

μPD7201A
MULTIPROTOCOL SERIAL COMMUNICATIONS CONTROLLER

October 1986

Description

The μPD7201A is a dual-channel multifunction peripheral communication controller (MPSCC) that satisfies a wide variety of serial data communication requirements in computer systems. Its basic function is as a serial-to-parallel, parallel-to-serial converter/controller, and it is software configurable for serial data communications applications.

The μPD7201A can handle asynchronous and synchronous byte-oriented protocols, such as IBM Bisync, and synchronous bit-oriented protocols such as HDLC and IBM SDLC. It also supports virtually any other serial protocol for applications other than data communications.

The μPD7201A can generate and check cyclic redundancy check (CRC) codes in any synchronous mode and can be programmed to check data integrity in various modes. The device also has facilities for modem controls in both channels. In applications where modem controls are not needed, they can be used for general-purpose I/O.

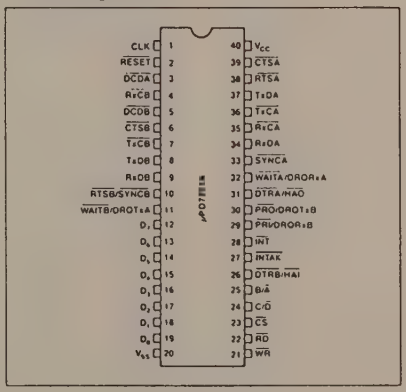
Features

- ☐ Two independent full duplex serial channels
- ☐ Four independent DMA channels for send/receive data for both serial inputs/outputs
- ☐ Programmable interrupt vectors and interrupt priorities
- ☐ Modem control signals
- ☐ Variable software programmable data rate, up to 1 Mbaud at 5 MHz clock
- ☐ Double buffered transmitter data and quadruply buffered receive data
- ☐ Selectable CRC algorithm
- ☐ Selection of interrupt, DMA or polling mode of operation
- ☐ Asynchronous operation
  - — Character length: ×1, ×16, ×32, or ×64 lock frequency
  - — Parity: odd, even, or disable
  - — Break generation and detection
  - — Interrupt on parity, overrun, or framing errors
- ☐ Monosync, bisync, and external sync operations
  - — Software selectable sync characters
  - — Automatic sync insertion
  - — CRC generation and checking
- ☐ HDLC and SDLC operations
  - — Abort sequence generation and detection
  - — Automatic zero insertion and detection
  - — Address field recognition
  - — CRC generation and checking
  - — I-field residue handling
- ☐ N-channel MOS technology
- ☐ Single +5V power supply; interface to most microprocessors including 8080, 8085, 8086, and others.
- ☐ Single-phase TTL clock
- ☐ Plastic and ceramic dual-in-line packages

Ordering Information

Part Number	Package Type
μPD7201AC	40-pin plastic DIP
μPD7201AD	40-pin ceramic DIP

Pin Configuration



Pin Identification

No.	Symbol	Function
1	CLK	System clock input
2	$\overline{RESET}$	Reset input
3	$\overline{DCDA}$	Data carrier detect input A
4	$\overline{RxCB}$	Receiver clock input B
5	$\overline{DCDB}$	Data carrier detect input B
6	$\overline{CTSB}$	Clear to send input B
7	$\overline{TxCB}$	Transmitter clock input B
8	TxDB	Transmit data output B
9	RxDB	Receive data input B
10	$\overline{RTSB}/\overline{SYNCB}$	Request to send output B/Synchronization input/output B
11	$\overline{WAITB}/DRQTxA$	Wait output B/Transmit DMA request output A
12–19	D_7-D_0	Data Bus
20	V_{SS}	Ground
21	$\overline{WR}$	Write strobe input
22	$\overline{RD}$	Read strobe input
23	$\overline{CS}$	Chip select input
24	C/$\overline{D}$	Control/data input
25	B/$\overline{A}$	Channel select input
26	$\overline{DTRB}/HAI$	Data terminal output B/Hold acknowledge input
27	$\overline{INTAK}$	Interrupt acknowledge input
28	$\overline{INT}$	Interrupt request output
29	$\overline{PRI}/DRQRxB$	Interrupt priority input/Receive DMA request output B
30	$\overline{PRO}/DRQTxB$	Interrupt priority output/Transmit DMA request output B
31	$\overline{DTRA}/HAO$	Data terminal output A/Hold acknowledge output
32	$\overline{WAITA}/DRQRxA$	Wait output A/Receive DMA request output A
33	$\overline{SYNCA}$	Synchronization input/output A
34	RxDA	Receive data input A
35	$\overline{RxCA}$	Receiver clock input A
36	$\overline{TxCA}$	Transmitter clock input A
37	TxDA	Transmit data output A
38	$\overline{RTSA}$	Request to send output A
39	$\overline{CTSA}$	Clear to send input A
40	V_{CC}	+5 V

Pin Functions

CLK [System Clock]

A TTL-level clock signal is applied to the CLK input. The system clock frequency must be at least 4.5 times the data rate.

$\overline{RESET}$ [Reset]

A low on the $\overline{RESET}$ input (one complete CLK cycle minimum) initializes the MPSCC to the following conditions: receivers and transmitters disabled, TxDA and TxDB set to marking (high), and modem controls ($\overline{DTRA}$, $\overline{DTRB}$, $\overline{RTSA}$, $\overline{RTSB}$) set high.

In addition, all interrupts are disabled and all interrupt and DMA requests are cleared. All control registers must be rewritten after a reset before transmission or reception can be restarted.

$\overline{DCDA}$, $\overline{DCDB}$ [Data Carrier Detect]

The $\overline{DCDA}$ and $\overline{DCDB}$ inputs go low to indicate the presence of valid serial data at RxD. The MPSCC may be programmed so that the receiver is enabled only when $\overline{DCD}$ is low, and so that any change in state that lasts longer than the minium specified pulse width causes an interrupt and latches the $\overline{DCD}$ status bit to the new state.

$\overline{RxCA}$, $\overline{RxCB}$ [Receiver Clock]

The $\overline{RxCA}$ and $\overline{RxCB}$ inputs control sampling and shifting serial data at RxDA and RxDB. The MPSCC can be programmed so that the clock rate is 1, 16, 32, or 64 times the data rate. RxD is sampled on the rising edge of $\overline{RxC}$. $\overline{RxC}$ features a Schmitt-trigger input for relaxed rise and fall time requirements.

$\overline{TxCA}$, $\overline{TxCB}$ [Transmitter Clock]

The $\overline{TxCA}$ and $\overline{TxCB}$ inputs control the rate at which data is shifted out at TxDA and TxDB. The MPSCC can be programmed so that the clock rate is 1, 16, 32, or 64 times the data rate. Data changes on the falling edge of $\overline{TxC}$. $\overline{TxC}$ features a Schmitt-trigger input for relaxed rise and fall time requirements.

TxDA, TxDB [Transmit Data]

TxDA and TxDB output serial data from the MPSCC. (Marking high).

RxDA, RxDB [Receive Data]

RxDA and RxDB input serial data to the MPSCC. (Marking high.)

$\overline{CTSA}$, $\overline{CTSB}$ [Clear to Send]

The $\overline{CTSA}$ and $\overline{CTSB}$ inputs go low to indicate that the receiving modem or peripheral is ready to receive data from the MPSCC. The MPSCC can be programmed so that the transmitter is enabled only when CTS is low. As with $\overline{DCD}$, the MPSCC can be programmed to cause an interrupt and latch the new state when $\overline{CTS}$ changes state for longer than the minimum specified pulse width.

$\overline{RTSA}$, $\overline{RTSB}$ [Request to Send]

When the MPSCC is in one of the synchronous modes, $\overline{RTSA}$ and $\overline{RTSB}$ are general-purpose outputs that can be set or reset with commands to the MPSCC. In asynchronous mode, $\overline{RTS}$ is active (low) as soon as it is programmed on. However, when programmed off, $\overline{RTS}$ remains active until the transmitter is completely empty. This feature simplifies the programming required to perform modem control.

$\overline{SYNCA}$, $\overline{SYNCB}$ [Synchronization]

The function of the $\overline{SYNCA}$ and $\overline{SYNCB}$ pins depends on the MPSCC operating mode. In asynchronous mode, $\overline{SYNC}$ is used as an input that the processor can read. It can be programmed to generate an interrupt in the same manner as $\overline{DCD}$ or $\overline{CTS}$.

In external sync mode, $\overline{SYNC}$ is an active-low input that notifies the MPSCC that synchronization has been achieved (see timing waveforms for details). Once synchronization is achieved, $\overline{SYNC}$ should be held low until synchronization is lost or a new message is about to start.

In internal synchronization modes (monosync, bisync, HDLC), $\overline{SYNC}$ is an output which is active (low) whenever a SYNC character match is made. There is no qualifying logic associated with this function. Regardless of character boundaries, $\overline{SYNC}$ is active on any match.

DRQTxA, DRQTxB, DRQRxA, DRQRxB [DMA Request]

When a DRQTxA, DRQTxB, DRQRxA, or DRQRxB output is active (high), it indicates to a DMA controller that a transmitter or receiver is requesting a DMA data transfer.

$\overline{WAITA}$, $\overline{WAITB}$ [Wait]

The $\overline{WAITA}$ and $\overline{WAITB}$ outputs synchronize the processor with the MPSCC when block transfer mode is used. It can be programmed to operate with either the receiver or transmitter, but not both simultaneously. $\overline{WAIT}$ is normally inactive (high). If the processor tries, for example, to perform an inappropriate data transfer such as a write to the transmitter when the transmitter buffer is full, the $\overline{WAIT}$ output for the channel will go active (low) until the MPSCC is ready to accept the data. The $\overline{CS}$, C/$\overline{D}$, B/$\overline{A}$, $\overline{RD}$, and $\overline{WR}$ inputs must remain stable while wait is active. (Open drain.)

D_0-D_7 [Data Bus]

The three-state data bus lines are connected to the system data bus. Data or status from the MPSCC is output on these lines when $\overline{CS}$ and $\overline{RD}$ are active (low). Data and commands are latched into the MPSCC on the rising edge of $\overline{WR}$ when $\overline{CS}$ is active.

$\overline{WR}$ [Write Strobe]

A low on the $\overline{WR}$ input (with either CS during the read cycle or HAI during a DMA cycle) notifies the MPSCC to write data or control information to the device.

$\overline{RD}$ [Read Strobe]

A low on the $\overline{RD}$ input (with either $\overline{CS}$ during a read cycle or $\overline{HAI}$ during a DMA cycle) notifies the MPSCC to read data or status from the device.

$\overline{CS}$ [Chip Select]

A low on the $\overline{CS}$ input allows the MPSCC to transfer data or commands during a read or write cycle.

C/$\overline{D}$ [Control/Data]

The C/$\overline{D}$ input, with $\overline{RD}$, $\overline{WR}$, $\overline{CS}$, and B/$\overline{A}$ selects the data register (C/$\overline{D}$ = 0) or the control and status registers (C/$\overline{D}$ = 1) for access over the data bus.

B/$\overline{A}$ [Channel Select]

B/$\overline{A}$ input low selects channel A and B/$\overline{A}$ high selects channel B for access during a read or write cycle.

$\overline{DTRA}$, $\overline{DTRB}$ [Data Terminal]

The $\overline{DTRA}$ and $\overline{DTRB}$ outputs are general-purpose, active-low outputs which may be set or reset with commands to the MPSCC.

$\overline{\text{INTAK}}$ [Interrupt Acknowledge]

The processor generates two or three $\overline{\text{INTAK}}$ low pulses (depending on the processor type) to signal all peripheral devices that an interrupt acknowledge sequence is taking place. During the interrupt acknowledge sequence, the MPSCC, if so programmed, places information on the data bus to vector the processor to the appropriate interrupt service location.

$\overline{\text{INT}}$ [Interrupt Request]

The $\overline{\text{INT}}$ output is pulled low when an internal interrupt request is accepted. (Open drain.)

$\overline{\text{PRI}}$ [Interrupt Priority In]

The $\overline{\text{PRI}}$ input informs the MPSCC that the highest priority device is requesting an interrupt. It is used with $\overline{\text{PRO}}$ to implement a priority-resolution daisychain when there is more than one interrupting device. The state of $\overline{\text{PRI}}$ and the programmed interrupt mode determine the MPSCC's response to an interrupt acknowledge sequence.

$\overline{\text{PRO}}$ [Interrupt Priority Out]

The $\overline{\text{PRO}}$ output is active (low) when $\overline{\text{PRI}}$ is active (low) and the MPSCC is not requesting an interrupt ($\overline{\text{INT}}$ is not active).

The active state informs the next lower priority device that there are no higher priority interrupt requests pending during an acknowledge sequence.

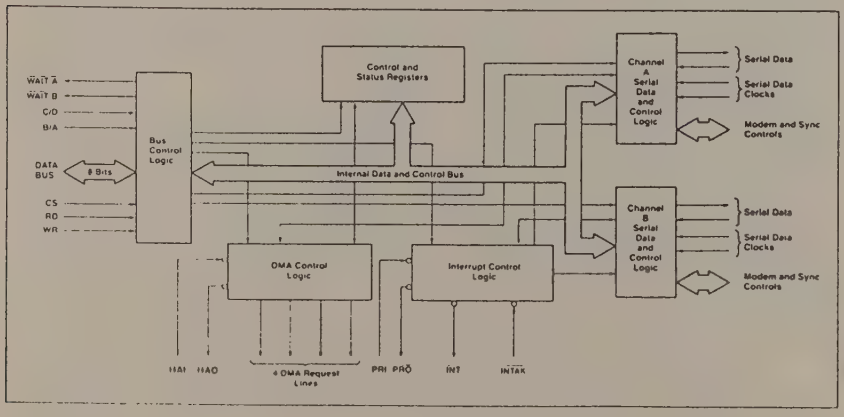
$\overline{\text{HAI}}$ [Hold Acknowledge In]

The $\overline{\text{HAI}}$ input goes low to notify the MPSCC that the host processor has acknowledged the DMA request and has placed itself in the hold state. The MPSCC then performs a DMA cycle for the highest priority outstanding DMA request, if any.

$\overline{\text{HAO}}$ [Hold Acknowledge Out]

The $\overline{\text{HAO}}$ output, with $\overline{\text{HAI}}$, implements a priority-resolution daisychain for multiple DMA devices. $\overline{\text{HAO}}$ is active (low) when $\overline{\text{HAI}}$ is active and there are no DMA requests pending in the MPSCC.

Block Diagram



Absolute Maximum Ratings

T_A = 25°C

Power Supply, V_{CC}	−0.5 to + 7.0 V
Input Voltage, V_I	−0.5 to + 7.0 V
Output Voltage, V_O	−0.5 to + 7.0 V
Operating temperature, T_{OPT}	0°C to + 70°C
Storage Temperature, T_{STG}	−65°C to + 150°C

Comment: Exposing the device to stresses above those listed in Absolute Maximum Ratings could cause permanent damage. The device is not meant to be operated under conditions outside the limits described in the operational section of this specification. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Capacitance

T_A = 25°C; V_{CC} = GND = 0V

		Limits			
Parameter	Symbol	Min	Max	Unit	Test Conditions
Input Capacitance	C_{IN}		10	pF	fc = 1MHz
Output Capacitance	C_{OUT}		15	pF	Unmeasured pins
I/O Capacitance	$C_{I/O}$		20	pF	returned to GND.

DC Characteristics

T_A = 0°C to + 70°C; V_{CC} = +5 V ± 10%

		Limits			
Parameter	Symbol	Min	Max	Unit	Test Conditions
Input low voltage	V_{IL}	−0.5	+0.8	V	
Input high voltage	V_{IH}	+2.0	V_{CC}+0.5	V	
Output low voltage	V_{OL}		+0.45	V	I_{OL} = +2.0 mA
Output high Voltage	V_{OH}	+2.4		V	I_{OH} = 200μA
Input leakage current	I_{IL}		±10	μA	V_{IN}=V_{CC} to 0 V
Output leakage current	I_{OL}		±10	μA	V_{OUT}=V_{CC} to 0 V
V_{CC} supply current	I_{CC}		230	mA	

AC Characteristics

T_A = 0°C to + 70°C; V_{CC} = +5 V ± 10%

		Limits			
Parameter	Symbol	Min	Max	Unit	Test Conditions
Clock cycle	t_{CY}	200	4000	ns	
Clock high width	t_{CH}	70	2000	ns	
Clock low width	t_{CL}	70	2000	ns	

AC Characteristics (cont)

T_A = 0°C to + 70°C, V_{CC} = +5 V ± 10%

		Limits			
Parameter	Symbol	Min	Max	Unit	Test Conditions
Clock rise time	t_r	0	30	ns	
Clock fall time	t_f	0	30	ns	
Address setup to $\overline{RD}$	t_{AR}	0		ns	
Address hold from $\overline{RD}$	t_{RA}	0		ns	
$\overline{RD}$ pulse width	t_{RR}	200		ns	
Data output delay from address	t_{AD}		200	ns	
Data output delay from $\overline{RD}$	t_{RD}		200	ns	
Data float delay from $\overline{RD}$	t_{DF}	10	100	ns	
Address setup from $\overline{WR}$	t_{AW}	0		ns	
Address hold from $\overline{WR}$	t_{WA}	0		ns	
$\overline{WR}$ pulse width	t_{WW}	200		ns	
Data setup to $\overline{WR}$	t_{DW}	130		ns	
Data hold from $\overline{WR}$	t_{WD}	0		ns	
PRO delay from $\overline{PRI}$	t_{PIPO}		100	ns	
PRO delay from $\overline{INTAK}$	t_{IAPO}		200	ns	
$\overline{PRI}$ setup to $\overline{INTAK}$	t_{PHA}	0		ns	
$\overline{PRI}$ hold from $\overline{INTAK}$	t_{IAPI}	20		ns	
$\overline{INTAK}$ pulse width	t_{IAIA}	200		ns	
Data output delay from $\overline{INTAK}$	t_{IAD}		200	ns	
Data float delay from $\overline{INTAK}$	t_{DF}	10	100	ns	
Request hold from $\overline{RD}$/$\overline{WR}$	t_{CO}		150	ns	
$\overline{HAI}$ setup to $\overline{RD}$/$\overline{WR}$	t_{HIC}	300		ns	
$\overline{HAI}$ hold from $\overline{RD}$/$\overline{WR}$	t_{CHI}	0		ns	
HAO delay from $\overline{HAI}$	t_{HIHO}		100	ns	
Data clock cycle	t_{DCY}	400		ns	RxC, TxC

AC Characteristics (cont)

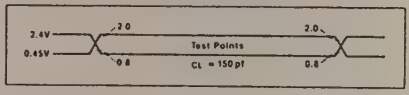
T_A = 0°C to + 70°C; V_{CC} = +5 V ± 10%

Parameter	Symbol	Limits Min	Limits Max	Unit	Test Conditions
Data clock high width	t_{DCH}	180		ns	RxC, TxC
Data clock low width	t_{DCL}	180		ns	RxC, TxC
Tx data delay from TxC	t_{TCTD}		300	ns	x1 Mode
			1000	ns	x16, x32, x64 Mode
Rx data setup to RxC	t_{RDRC}	0		ns	
Rx Data hold from RxC	t_{RCRD}	140		ns	
INT delay Time from Tx Data	t_{IDI}		4-6	t_{CY}	
INT delay Time from RxC	t_{RCI}		7-11	t_{CY}	
CTS, DCD, SYNC high pulse width	t_{MH}	200		ns	
CTS, DCD, SYNC low pulse width	t_{ML}	200		ns	
External INT from CTS, DCD, SYNC	t_{MF}		500	ns	
Recovery time between controls	t_{RV}	300		ns	
WAIT delay time from Address	t_{AWT}		120	ns	
SYNC setup to RxC	t_{RCS}		100	ns	

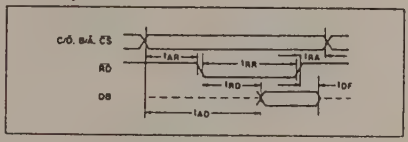
Note: 1. RESET must be active for a minimum of one complete CLK cycle.
2. In all modes system clock rate must be 4.5 times data rate.

Timing Waveforms

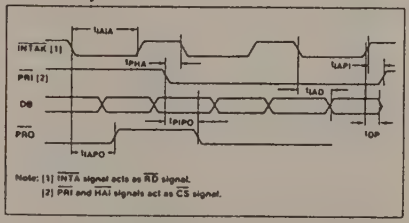
AC Waveform Measurement Points



Read Cycle

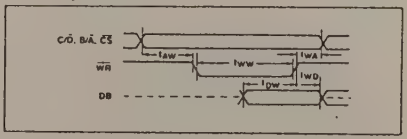


INTAK Cycle

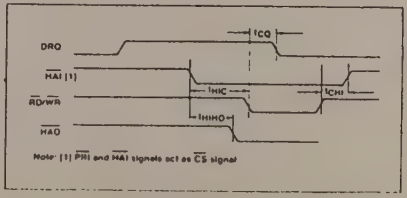


Note: (1) INTA signal acts as RD signal.
(2) PRI and HAI signals act as CS signal.

Write Cycle



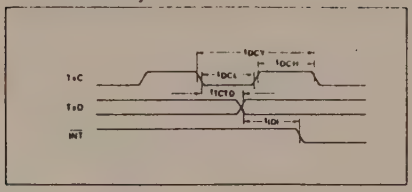
DMA Cycle



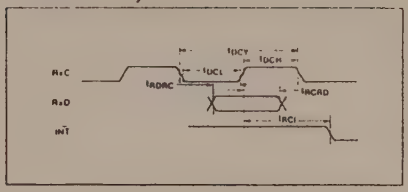
Note: (1) PRI and HAI signals act as CS signal

Timing Waveforms (cont)

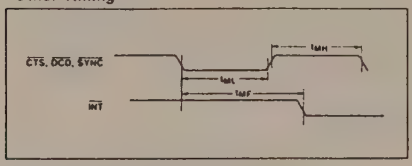
Transmit Data Cycle



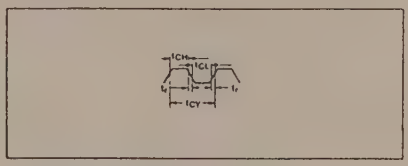
Receive Data Cycle



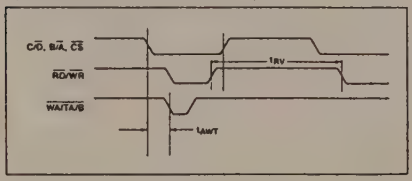
Other Timing



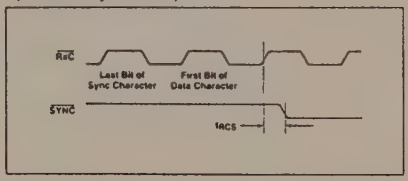
Clock



Read/Write Cycle
(Software Block Transfer Mode)



Sync Pulse Generation
(External Sync Mode)



Programming the MPSCC

Software operation of the MPSCC includes consistent register organization and high-level command structure to help minimize the number of operations required to implement complex protocol designs. The MPSCC also has extensive interrupt and status reporting capabilities to simplify programming.

The MPSCC Registers

The MPSCC interfaces to the system software with a number of control and status registers associated with each channel (see tables 1 and 2). Commonly used commands and status bits are accessed directly through control and status register 0. Other functions are accessed indirectly with a register pointer to minimize the address space that must be dedicated to the MPSCC.

All control and status registers except CR2 are separately maintained for each channel. Control and status register 2 are linked with the overall operation of the MPSCC and have different meanings when addressed through different channels.

Before initializing the MPSCC, first program control register 2A (2B if desired) to establish the MPSCC processor/bus interface mode. Each channel may then be programmed for separate use beginning with control register 4 to set the protocol mode for that channel. The remaining registers may then be programmed in any order.

Table 1. Control Registers

Control Register	Function
0	Frequently used commands and register pointer control
1	Interrupt control
2	Processor/bus interface control
3	Receiver control
4	Mode control
5	Transmitter control
6	Sync/address character
7	Sync character

Control Register 0

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
CRC Control Command		Command			Register Pointer		

Register Pointer [D_0·D_2]

The register pointer specifies which register number is accessed at the next control register write or status register read. After a hardware or software reset, the register pointer is set to zero. Therefore, the first control byte goes to control register 0. When the register pointer is set to a value other than zero the next control or status ($C/\overline{D} = 1$) access is to the specified register. The pointer is then reset to 0 by setting the register pointer.

Commands [D_3·D_5]

Commands commonly used during the operation of the MPSCC are grouped in control register 0. They include the following:

Null [000]: This command has no effect and is used only to set the register pointer or issue a CRC command.

Send Abort [001]: When operating in the HDLC mode, this command causes the MPSCC to transmit the HDLC abort code by issuing 8 to 13 consecutive 1s. Any data currently in the transmitter or the transmitter buffer is destroyed. After sending the abort, the transmitter reverts to the idle phase (flags). When using the Tx byte count mode enable (D_6 of CR1), the send abort command is automatically issued when an underrun condition occurs.

Table 2. Status Registers

Status Register	Function
0	Buffer and "external/status" status
1	Received character error and special condition status
2 (Channel B only)	Interrupt Vector
3	Tx byte count register, low byte
4	Tx byte count register, high byte

Reset External Status Interrupt [010]: When the external/status change flag is set, the condition of bits D_3·D_7 of status register 0 are latched to capture the short pulses that may occur. The reset external/status interrupts command reenables the latches so that new interrupts may be sensed.

Channel Reset [011]: This command has the same effect on a single channel as an external reset at pin 2. A channel reset command to channel A rests the internal interrupt prioritization logic. This does not occur when a channel reset command is issued to channel B. All control registers associated with the channel to be reset must be reinitialized. After a channel reset, wait at least four system clock cycles before writing new commands or controls to that channel.

Enable Interrupt on Next Character [100]: Issue this command at any time when operating the MPSCC in an interrupt on first received character mode. This command must be issued at the end of a message to reenable the interrupt logic for the next received character (first character of the next message).

Reset Pending Transmitter Interrupt/DMA Request [101]: A pending transmitter buffer empty interrupt or DMA request can be reset without sending another character by issuing this command (typically at the end of a message). A new transmitter buffer empty interrupt or DMA request is not made until another character has been loaded and transferred to the transmitter shift register or when, if operating in synchronous mode, the first CRC character has been sent.

Error Reset [110]: This command resets a special receive condition interrupt. It also reenables the parity and overrun error latches that allow error checking at the end of a message.

End of Interrupt [111] [Channel A Only] : Once an interrupt request has been issued by the MPSCC, all lower priority internal and external interrupts in the daisy chain are held off to permit the current interrupt to be serviced while allowing higher priority interrupts to occur. At some point in the interrupt service routine (generally at the end), the end of the interrupt command must be issued to channel A to reenable the daisy chain and allow any pending lower priority internal interrupt requests to occur. The EOI command must be sent to channel A for interrupts that occured on either channel.

CRC Control Commands [D_6-D_7]

The following commands control the operation of the CRC generator/checker logic:

Null [00] : This command has no effect and is used when issuing other commands or setting the register pointer.

Reset Receiver CRC Checker [01] : This command resets the CRC checker to zero when the channel is in a synchronous mode. It resets to all 1s when in an HDLC mode.

Reset Transmitter CRC Generator [10] : This command resets the CRC generator to zero when the channel is in a synchronous mode. It resets to all 1s when in an HDLC mode.

Reset Idle/CRC Latch [11] : This command resets the idle/CRC latch so that when a transmitter underrun condition occurs (transmitter has no more characters to send), the transmitter enters the CRC phase of operation and begins to send the 16-bit CRC character calculated up to that point. The latch is then set so that if the underrun condition persists, idle characters are sent following the CRC. After a hardware or software reset, the latch is in the set state. This latch is automatically reset after the first character has been loaded into the Tx buffer in the HDLC mode.

Control Register 1

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Wait Function Enable	Tx Byte Count Mode Enable	Wait on Receive Transmitter	Receiver Interrupt Mode		Condition Affects Vector	Transmitter Interrupt Enable	Ext/Status INT Enable

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Low Byte							

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
High Byte							

External/Status Interrupt Enable [D_0]

When this bit is set to one, the MPSCC issues an interrupt whenever any of the following conditions occur:
- Transition of the DCD, CTS or SYNC input pin
- Entering or leaving synchronous hunt phase, break detection or termination
- HDLC abort detection or termination
- Idle/CRC latch set (CRC being sent)
- After ending flag is sent in the HDLC mode

Transmitter Interrupt Enable [D_1]

When this bit is set to one, the MPSCC issues an interrupt when the following conditions occur:
- A character currently in the transmitter buffer is transferred to the shift register (transmitter buffer becomes empty), or
- The transmitter enters the idle phase and begins transmitting sync or flag characters.
- The Tx byte count mode enable bit is set (D_6 of CR1 = 1). The 7201A will automatically issue a Tx interrupt or DMA request when the transmitter becomes enabled (D_3 of CR5 = 1).

Condition Affects Vector [D_2]

When this bit is set to zero, the fixed vector programmed in CR2B during MPSCC initialization is returned in an interrupt acknowledge sequence. When this bit is set to one, the vector is modified to reflect the condition that caused the interrupt. (Programmed in channel B for both channels).

Receiver Interrupt Mode [D_3 - D_4]

This field controls how the MPSCC interrupt/DMA logic handles the character received condition.

Receiver Interrupts/DMA Request Disabled [00] : The MPSCC does not issue an interrupt or a DMA request when a character has been received.

Interrupt/DMA on First Received Character Only [01] : In this mode the MPSCC issues an interrupt only for the first character received after an enable interrupt/DMA on first character command (CR0) has been given. If the channel is in a DMA mode, a DMA request is issued for each character received, including the first. In general, use this mode whenever the MPSCC is in a DMA or block transfer mode. This will signal the processor that the beginning of an incoming message has been received.

Interrupt [and Issue a DMA Request] on All Received Characters [10]: In this mode an interrupt (and DMA request if the DMA mode is selected) is issued whenever there is a character present in the receiver buffer. A parity error is considered a special receive condition.

Interrupt [and Issue a DMA Request] on All Received Characters [11]: This mode is the same as the one above, except that a parity error is not considered a special receive condition. The following are considered special receive conditions:
- Receive overrun error
- Asynchronous framing error
- Parity error (if specified)
- HDLC end of message (final flag received)

Wait on Receiver/Transmitter [D_5]

If the wait function is enabled for block mode transfers, setting this bit to zero causes the MPSCC to issue a wait (WAIT output goes low) when the processor attempts to write a character to the transmitter while the transmitter buffer is full. Setting this bit to one causes the MPSCC to issue a wait when the processor attempts to read a character from the receiver while the receiver buffer is empty.

Tx Byte Count Enable [D_6]

Each channel has a 16-bit Tx byte count register used for automatic transmit termination. When this bit is set to one, the next two consecutive command cycle writes will be to the byte count register. The first byte is loaded into the lower 8 bits and the second to the upper 8 bits of the byte count register. The byte count register holds the number of transfers to be performed by the transmitter. A byte counter is incremented each time a transfer is performed until the value of the byte counter is equal to the value in the byte count register. When equal, interrupts or DMA requests will be stopped until the byte count enable bit is issued and a new byte count is loaded into the byte count register. If a transmit underrun occurs in the HDLC mode, and the byte count is not equal to the byte count register, an abort sequence will be sent automatically.

Also, when using the Tx byte count mode, a transmit interrupt or DMA request will automatically become active after issuing the TX enable command to CR5.

The Tx byte count mode can be cleared by either a channel reset command or a hardware reset.

Wait Function Enable [D_7]

Setting this bit to one enables the wait function selected by D_5 of CR1.

Control Register 2 (Channel A)

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Pin 10 SYNCB/ RTSB	Rx INT Mask	Interrupt Vector			Priority	DMA Mode Select	

DMA Mode Select [$D_0 \cdot D_1$]

Setting this field determines whether channel A or B is used in a DMA mode [data transfers are performed by a DMA controller], or in a non-DMA mode where transfers are performed by the processor in either a polled, interrupt, or block transfer mode. The functions of some MPSCC pins are also controlled by this field. See table 3.

Priority [D_2]

This bit selects the relative priorities of the various interrupt and DMA conditions according to the application requirements. See table 4.

Interrupt Vector Mode [$D_3 \cdot D_5$]

This field determines how the MPSCC responds to an interrupt acknowledge sequence from the processor. See table 5.

Rx INT Mask [D_6]

This option is generally used in the DMA modes. Enabling this bit inhibits the interrupt from occuring when the interrupt/DMA request on first received character mode is selected. In other words, only a DMA request will be generated when the first character is received.

Table 3. DMA Mode Selection

D_1	D_0	Channel A	Channel B	11	26	29	30	31	32
					Pin Function				
0	0	Non-DMA	Non-DMA	WAITB	DTRB	PRI	PRO	DTRA	WAITA
0	1	DMA	Non-DMA	DRQTxA	HAI	PRI	PRO	HAO	DRQRxA
1	0	DMA	DMA	DRQTXA	HAI	DRQRxB	DRQTxB	HAO	DRQRxA
1	1	DMA	DMA	DRQTxA	DTRB	DRQRxB	DRQTxB	DTRA	DRQRxA

Table 4. DMA/Interrupt Priorities

D_2	Mode Channel A	Mode Channel B	DMA Priority Relation	Interrupt Priority Relation
0	INT	INT		SRxA, RxA > TxA > SRxB, RxB > TxB > ExTA > ExTB
1	INT	INT		SRxA, RxA > SRxB, RxB > TxA > TxB > ExTA > ExTB
0	DMA	INT	RxA > TxA	SRxA, RxA > SRxB, RxB > TxB > ExTA > ExTB
1	DMA	INT	RxA > TxA	SRxA, RxA > SRxB, RxB > TxB > ExTA > ExTB
0	DMA	DMA	RxA > TxA > RxB > TxB	SRxA, RxA > SRxB, RxB > TxB > ExTB
1	DMA	DMA	RxA > RxB > TxA > TxB	SRxA, RxA > SRxB, RxB > ExTA, ExTB

Table 5. Interrupt Acknowledge Sequence Response

D_5	D_4	D_3	Mode	Status Register 2B and Interrupt Vector Bits Affected When Condition Affects Vector is Enabled
0	0	0	Nonvectored	$D_4D_3D_2$
0	0	1	Nonvectored	$D_4D_3D_2$
0	1	0	Nonvectored	$D_2D_1D_0$
0	1	1	Illegal	
1	0	0	8085 Master	$D_4D_3D_2$
1	0	1	8085 Slave	$D_4D_3D_2$
1	1	0	8086	$D_2D_1D_0$
1	1	1	8085/8259A Slave	$D_4D_3D_2$

Pin 10 $\overline{\text{SYNCB}}/\overline{\text{RTSB}}$ Select [D_7]

Programming a zero into this bit selects RTSB as the function of pin 10. A one selects SYNCB as the function.

Control Register 2 (Channel B)

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Interrupt Vector							

Interrupt Vector [D_0 - D_7]

When using the MPSCC in the vectored interrupt mode, the contents of this register are placed on the bus during the appropriate portion of the interrupt acknowledge sequence. Its value is modified if status affects vector is enabled. The value of SR2B can be read at anytime. This feature is useful in determining the cause of an interrupt when using the MPSCC in a nonvectored interrupt mode.

Control Register 3

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Number of Received Bits per Character		Auto Enables	Enter Hunt Phase	Receiver CRC Enable	Address Search Mode	Sync Character Load Inhibit	Receiver Enable

Receiver Enable [D_0]

Setting this bit to one after the channel has been completely initialized allows the receiver to begin operation. This bit may be set to zero at any time to disable the receiver.

Sync Character Load Inhibit [D_1]

In the character synchronous modes, this bit inhibits the transfer of sync characters to the receiver buffer, thus performing a "sync-stripping" operation. When using the MPSCC's CRC checking ability, use this feature only to strip leading sync characters preceding a message, since the load inhibit does not exclude sync characters embedded in the message from the CRC calculation. Synchronous protocols using other types of block checking such as checksum or LRC are free to strip embedded sync characters.

Address Search Mode [D_2]

In the HDLC mode, setting this bit places the MPSCC in an address search mode. Character assembly does not begin until the 8-bit character (secondary address field) following the starting flag of a message matches either the address programmed into CR6 or the global address 11111111.

Receiver CRC Enable [D_3]

This bit enables and disables (1 = enable) the CRC checker in the character oriented protocol mode, allowing characters from the CRC calculation to be selectively included or excluded. The MPSCC has a one-character delay between the receiver shift register and the CRC checker so that the enabling or disabling takes affect with the last character transferred from the shift register to the receiver buffer. Therefore, there is one full character time in which to read the character and decide whether or not it should be included in the CRC calculation. In the HDLC mode, there is no 8-bit delay.

Enter Hunt Phase [D_4]

Although the MPSCC receiver automatically enters the sync hunt phase after a reset, there are other times when reentry is appropriate. This may occur when synchronization has been lost or, in an HDLC mode, to ignore the current incoming message. A one in this bit position at any time after initialization causes the MPSCC to reeneter the hunt phase.

Auto Enables [D_5]

Setting this bit to one causes the DCD and CTS inputs to act as enable inputs to the receiver and transmitter, respectively.

Number of Received Bits per Character [D_6 - D_7]

This field specifies the number of data bits assembled to make each character. The value may be changed while a character is being assembled and, if the change is made before the new number of bits has been reached, it affects that character. Otherwise, the new specifications take effect on the next character received. See table 6.

Control Register 4

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Clock Rate		Sync Mode		Number of Stop Bits per Sync Mode		Parity Even/ Odd	Parity Enable

Parity Enable [D_0]

Setting this bit to one adds an extra data bit containing parity information to each transmitted character. Each received character is expected to contain this extra bit, and the receiver parity checker is enabled.

Table 6. Received Bits per Character

D_7	D_6	Bits per Character
0	0	5
0	1	7
1	0	6
1	1	8

Table 7. Stop Bits

D_3	D_2	Mode
0	0	Synchronous modes
0	1	Asynchronous 1 bit time (1 stop bit)
1	0	Asynchronous 1½ bit times (1½ stop bits)
1	1	Asynchronous 2 bit times (2 stop bits)

Parity Even/Odd [D_1]

Programming a zero into this bit when parity is enabled selects odd parity for the received character. Conversely, a one in this bit selects even parity generation and checking.

Number of Stop Bits or Sync Mode [D_2 - D_3]

This field specifies whether the channel is used in a synchronous or an asynchronous mode. In an asynchronous mode, this field also specifies the number of bit times used as the stop bit length by the transmitter. The receiver always checks for one stop bit. See table 7.

Sync Mode [D_4 - D_5]

When the stop bits/sync mode field is programmed for synchronous modes (D_2, D_3 = 00), this field specifies the particular synchronous format to be used. This field is ignored in an asynchronous mode. See table 8.

Clock Rate [D_6 - D_7]

This field specifies the relationship between the transmitter and receiver clock inputs (TxC, RxC) and the actual data rates at TxD and RxD. When operating in a synchronous mode, a 1x clock rate must be specified. In asynchronous modes, any of the rates may be specified. However, with a 1x clock rate, the receiver cannot determine the center of the start bit. In this mode, the sampling (rising) edge of RxC must be externally synchronized with the data. See table 9.

Table 8. Synchronous Formats

Sync Mode 1 D_5	Sync Mode 2 D_4	Mode
0	0	8-bit internal synchronization character (monosync)
0	1	16-bit internal synchronization character (bisync)
1	0	SDLC/HDLC
1	1	External synchronization (SYNC pin becomes an input)

Table 9. Clock Rates

Clock Rate 1 D_7	Clock Rate 2 D_6	Clock Rate
0	0	Clock Rate = 1x Data Rate
0	1	Clock Rate = 16x Data Rate
1	0	Clock Rate = 32x Data Rate
1	1	Clock Rate = 64x Data Rate

Control Register 5

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
DTR	Number of Transmitted Bits per Character		Send Break	Transmitter Enable	CRC Polynomial Select	RTS	Transmitter CRC Enable

Transmitter CRC Enable [D_0]

A one or a zero enables or disables (respectively) the CRC generator calculation. The enable or disable does not take effect until the next character is transferred from the transmitter buffer to the shift register, thus allowing specific characters to be included or excluded from the CRC calculation. By setting or resetting this bit just before loading the next character, it and subsequent characters are included or excluded from the calculation. If this bit is zero when the transmitter becomes empty, the MPSCC goes to the idle phase regardless of the state of the idle/CRC latch.

$\overline{RTS}$ [D_1]

In synchronous and HDLC modes, setting this bit to one causes the $\overline{RTS}$ pin to go low, while a zero causes it to go high. In an asynchronous mode, setting this bit to zero causes the $\overline{RTS}$ pin to go high when the transmitter is completely empty. This feature facilitates programming the MPSCC for use with asynchronous modems.

CRC Polynomial Select [D_2]

This bit selects the polynomial used by the transmitter and receiver for CRC generation and checking. A one selects the CRC-16 polynomial ($X^{16} + X^{15} + X^2 + 1$). A zero selects the CRC-CCITT polynomial ($X^{16} + X^{12} + X^5 + 1$). In an HDLC mode CRC-CCITT must be selected. Either polynomial may be used in other synchronous modes.

Transmitter Enable [D_3]

After a reset, the transmitted data output (TxD) is held high (marking) and the transmitter is disabled until this bit is set.

In an asynchronous mode TxD remains high until data is loaded for transmission.

When the transmitter is disabled in an asynchronous mode, any character currently being sent is completed before TxD returns to the marking state.

If the transmitter is disabled during the data phase in a synchronous mode, the current character is sent. TxD then goes high (marking). In an HDLC mode, the current character is sent, but the following marking line is zero-inserted. That is, the line goes low for one bit time out of every five.

Never disable the transmitter during the HDLC data phase unless a reset follows immediately. In either case, any character in the buffer register is held.

Disabling the transmitter during the CRC phase causes the remainder of the CRC character to be bit-substituted with the sync (or flag). The total number of bits transmitted is correct and TxD goes high after they are sent.

If the transmitter is disabled during the idle phase, the remainder of the sync (flag) character is sent. TxD then goes high.

Send Break [D_4]

Setting this bit to one immediately forces the transmitter output (TxD) low (spacing). This function overrides the normal transmitter output and destroys any data being transmitted, although the transmitter is still in operation. Resetting this bit releases the transmitter output.

Transmitted Bits per Character [D_5 - D_6]

This field controls the number of data bits transmitted in each character. The number of bits per character may be changed by rewriting this field just before the first character is loaded. See table 10.

Normally each character is sent to the MPSCC right-justified and the unused bits are ignored. However, when sending five bits or less, the data should be formatted as shown below to inform the MPSCC of the precise number of bits to be sent. See table 11.

Table 10. Transmitted Bits per Character

Transmitted Bits per Character 1 D_6	Transmitted Bits per Character D_5	Bits per Character
0	0	5 or less (see below)
0	1	7
1	0	6
1	1	8

Table 11. Transmitted Bits per Character for 5 Characters or Less

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Number of Bits per Character
1	1	1	1	0	0	0	D_0	1
1	1	1	0	0	0	D_1	D_0	2
1	1	0	0	0	D_2	D_1	D_0	3
1	0	0	0	D_3	D_2	D_1	D_0	4
0	0	0	D_4	D_3	D_2	D_1	D_0	5

$\overline{DTR}$ [Data Terminal Ready] [D_7]

When this bit is one, the $\overline{DTR}$ output is low (active). When this bit is zero, $\overline{DTR}$ is high.

Control Register 6

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Sync Byte 1							

Sync Byte 1 [D_0 - D_7]

Sync byte 1 is used in the following modes:

Monosync	8-bit sync character transmitted during the idle phase
Bisync	Least significant (first) 8 bits of the 16-bit transmit and receive sync character
External Sync	Sync character transmitted during the idle phase
HDLC	Secondary address value matched to secondary address field of the HDLC frame when the MPSCC is in the address search mode

Control Register 7

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Sync Byte 2							

Sync Byte 2 [D_0 - D_7]

Sync byte 2 is used in the following modes:

Monosync	8-bit sync character matched by the receiver
Bisync	Most significant (second) 8 bits of the 16-bit transmit and receive sync characters
HDLC	The flag character 01111110 must be programmed into control register 7 for flag matching by the MPSCC receiver

Status Register 0

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Break/ Abort	Idle/ CRC	CTS	Sync Status	DCD	Tx Buffer Empty	INT Pending	Rec'd Char Available

Received Character Available [D_0]

When this bit is set, it indicates that one or more characters in the receiver buffer are available for the processor to read. Once the processor has read all the available characters, the MPSCC resets this bit until a new character is received.

Interrupt Pending [D_1 - Channel A Only]

The interrupt pending bit is used with the interrupt vector register (status register 2) to make it easier to determine the MPSCC's interrupt status. This is useful in a nonvectored interrupt mode where the processor must poll each device to determine the interrupt source. In this mode, interrupt pending is set when status register 2B is read, the PRI input is active (low), and the MPSCC requests interrupt service.

It is not necessary to read the status registers of both channels to determine if an interrupt is pending. If the status affects vector is enabled and the interrupt pending is set, the vector read from SR2 contains valid condition information.

In a vectored interrupt mode, interrupt pending is set during the interrupt acknowledge cycle (on the leading edge of the second INTAK pulse) when the MPSCC is the highest priority device requesting interrupt service (PRI is active). In either mode, if there are no other pending interrupt requests, interrupt pending is reset when the end of the interrupt command is issued.

Transmitter Buffer Empty [D_2]

This bit is set whenever the transmitter buffer is empty — except during the transmission of CRC. The MPSCC uses the buffer to facilitate this function. After a reset, the buffer is considered empty and transmit buffer empty is set.

External/Status Flags [D_3 - D_7]

The following status bits reflect the state of the various conditions that cause an external/status interrupt. The MPSCC latches all external/status bits whenever a change occurs that would cause an external/status interrupt, regardless of whether this interrupt is enabled. This allows transient status changes on these lines to be saved.

When operating the MPSCC in an interrupt-driven mode for external/status interrupts, read status register 0 when this interrupt occurs and issue a reset external/status interrupt command to reenable the interrupt and the latches. To poll these bits without interrupts, issue the reset external/status interrupt command to first update the status to reflect the current values.

$\overline{DCD}$ [D_3] : This bit reflects the inverted state of the $\overline{DCD}$ input. When $\overline{DCD}$ is low the $\overline{DCD}$ status bit is high. Any transition on this bit causes an external/status interrupt request.

Sync Status [D_4] : The meaning of this bit depends on the operating mode of the MPSCC.

Asynchronous mode: Sync status reflects the inverted state of the $\overline{SYNC}$ input. When $\overline{SYNC}$ is low, sync status is high. Any transition on this bit causes an external/status interrupt request.

External synchronization mode: Sync status operates in the same manner as in asynchronous mode. The MPSCC's receiver synchronization logic is also tied to the sync status bit in an external synchronization mode. A low-to-high transition ($\overline{SYNC}$ input going low) informs the receiver that synchronization has started and character assembly begins.

A low-to-high transition on the $\overline{SYNC}$ input indicates that synchronization has been lost. The sync status becomes zero and an external/status is generated. The receiver remains in the receive data phase until the enter hunt phase bit in control register 3 is set.

Monosync, bisync, HDLC modes: In these modes, sync status indicates whether the MPSCC receiver is in the sync hunt or receive data phase of operation. A zero indicates that the MPSCC is in the receive data phase, and a one indicates that the MPSCC is in the sync hunt phase (as in after a reset or when the enter sync hunt bit sets to 1). As in the other modes, a transition on this bit causes an external/status interrupt. Note that entering a sync hunt phase (when programmed) or a reset causes an external/status interrupt request which may be cleared immediately with a reset external/status interrupt command.

$\overline{CTS}$ [D_5] : This bit reflects the inverted state of the $\overline{CTS}$ input. When $\overline{CTS}$ is low, the $\overline{CTS}$ status bit is high. Any transition on this bit causes an external/status interrupt request.

Idle/CRC [D_6] [Tx Underrun/EOM] : This bit indicates the state of the idle/CRC latch used in the synchronous mode. After a hardware reset, this bit is set to one, indicating that the transmitter is completely empty. When the MPSCC enters idle phase, it automatically transmits sync or flag characters.

In the HDLC mode, the MPSCC automatically resets this latch after the first byte of a frame is written to the Tx buffer.

When the transmitter is completely empty, the MPSCC sends the 16-bit CRC character and sets the latch again. An external/status interrupt is issued when the latch is set, indicating that CRC is being sent. No interrupt is issued when the latch is reset.

Break/Abort [D_7] : In the asynchronous mode, this bit indicates the detection of a break sequence (a null character plus framing error that occurs when the RxD input is held low, spacing, for more than one character time). Break/abort is reset when RxD returns high (marking).

In the HDLC mode, break/abort indicates the detection of an abort sequence when seven or more ones are received in sequence. It is reset when a zero is received.

Any transition of the break/abort bit causes an external/status interrupt.

Status Register 1

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
End of SDLC Frame	CRC Framing Error	Over-run Error	Parity Error	SDLC Residue Code			All Sent

All Sent [D_0]

This bit is set when the transmitter is empty and reset when a character is present in the transmitter buffer or shift register. This feature simplifies the mode control software routines. In the bit synchronous mode, this bit sets when the ending flag pattern is sent.

Residue Code [D_1 - D_3]

Since the data portion of an HDLC message can consist of any number of bits and not necessarily an integral number of characters, the MPSCC has special logic to determine and report when the end of frame flag has been received (that is, the boundary between the data field and the CRC character in the last few data characters that were just read).

When the end of frame condition is indicated (D_7 of status register 1 = 1) and there is a special receive condition interrupt (if enabled), the last bits of the CRC character are in the receiver buffer. The residue code for the frame is valid in the status register 1 byte associated with that data character. (SR1 tracks the received data in its own buffer).

The meaning of the residue code depends upon the number of bits per character specified for the receiver. The previous character refers to the last character read before the end of frame, and so on. See table 12.

Table 12. Residue Codes

8 Bits per Character				
D_3	D_2	D_1	Previous Character	2nd Previous Character
1	0	0	CCCCCCCC	CCCCCDDD
0	1	0	CCCCCCCC	CCCCDDDD
1	1	0	CCCCCCCC	CCCDDDDD
0	0	1	CCCCCCCC	CCDDDDDD
1	0	1	CCCCCCCC	CDDDDDDD
0	1	1	CCCCCCCC*	DDDDDDDD*
1	1	1	CCCCCCCD	DDDDDDDD
0	0	0	CCCCCCDD	DDDDDDDD

7 Bits per Character				
D_3	D_2	D_1	Previous Character	2nd Previous Character
1	0	0	CCCCCCC	CCCCCDD
0	1	0	CCCCCCC	CCCCDDD
1	1	0	CCCCCCC	CCCDDDD
0	0	1	CCCCCCC	CCDDDDD
1	0	1	CCCCCCC	CDDDDDD
0	1	1	CCCCCCC*	DDDDDDD*
0	0	0	CCCCCCD	DDDDDDD

6 Bits per Character				
D_3	D_2	D_1	Previous Character	2nd Previous
1	0	0	CCCCCC	CCCCCD
0	1	0	CCCCCC	CCCCCD
1	1	0	CCCCCC	CCCDDD
0	0	1	CCCCCC	CCDDDD
1	0	1	CCCCCC	CDDDDD
0	0	0	CCCCCC	DDDDDD

5 Bits per Character				
D_3	D_2	D_1	Previous Character	2nd Previous
1	0	0	CCCCC*	DDDD*
0	1	0	CCCCD	DDDDD
1	1	0	CCCDD	DDDDD
0	0	1	CCDDD	DDDDD
0	0	0	CDDDD	DDDDD

Notes: C = CRC bit
D = Valid data
* = No residue

Special Receive Condition Flags

The status bits described below—parity error (if parity as a special receive condition is enabled), receiver overrun error, CRC/framing error, and end of HDLC frame—all represent special receive conditions.

When any of these conditions occur and interrupts are enabled, the MPSCC issues an interrupt request. In addition, if a condition affect vector mode is enabled, the vector generated (and the contents of SR2B for nonvectored interrupts) is different from that of a received character available condition. Therefore, it is not necessary to analyze SR1 with each character to determine if an error has occurred.

Also, the parity error and receiver overrun error flags are latched. That is, once one of these errors occurs, the flag remains set for all subsequent characters until reset by the error reset command. Therefore read SR1 only at the end of a message to determine if either of these errors occurred anywhere in the message. The other flags are not latched and follow each character available in the receiver buffer.

Parity Error [D_4]: This bit is set and latched when parity is enabled and the received parity bit does not match the sense (odd or even) calculated from the data bits.

Receiver Overrun Error [D_5]: This error occurs and is latched when the receiver buffer already contains three characters and a fourth character is completely received, overwriting the last character in the buffer.

CRC/Framing Error [D_6]: In the asynchronous mode a framing error is flagged (but not latched) when no stop bit is detected at the end of a character (RxD is low one bit time after the center of the last data or parity bit). When this condition occurs, the MPSCC waits an additional one-half bit time before sampling again so that the framing error is not interpreted as a new start bit.

In the synchronous mode, this bit indicates the result of the comparison between the current CRC result and the appropriate check value. It is usually set to one, since a message rarely indicates a correct CRC result until correctly completed with the CRC check character. Note that a CRC error does not result in a special receive condition interrupt.

End of HDLC Frame [EOF] [D_7]: This status bit is used only in the bit synchronous mode to indicate that the end of frame flag has been received and that the CRC error flag and residue code are valid. This flag can be reset at any time by issuing an error reset command. The MPSCC also automatically resets this bit when the first character of the next message is sent.

Status Register 2B

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Interrupt Vector							

Interrupt Vector [D_0 - D_7 · Channel B Only]

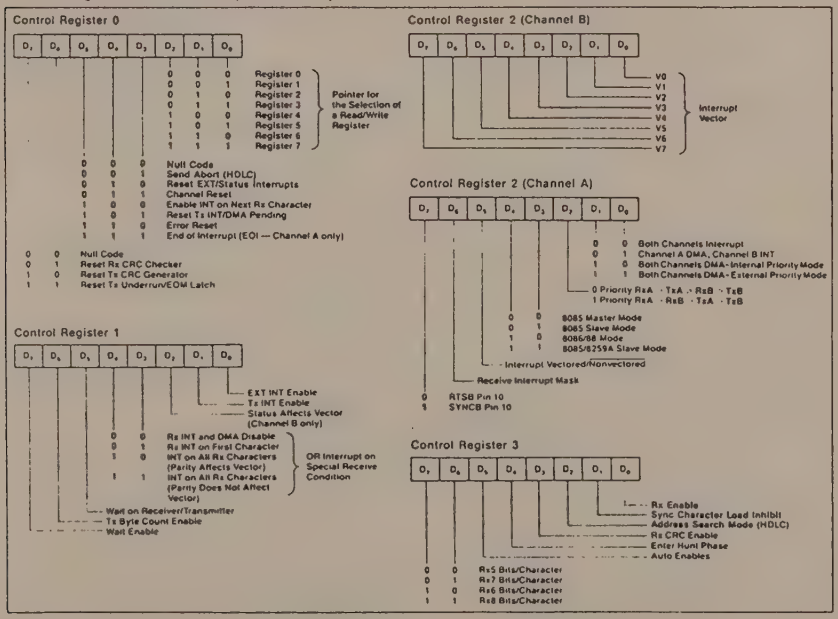
Reading status register 2B returns the interrupt vector that is programmed into control register 2B. If a condition affects vector mode is enabled, the value of the vector is modified as shown in table 13.

Code 111 can mean either channel A special receive condition or no interrupt pending. Examine the interrupt pending bit (D_1 of status register 0, channel A), to distinguish which it means. In a nonvectored interrupt mode, the vector register must be read first for the interrupt pending to be valid.

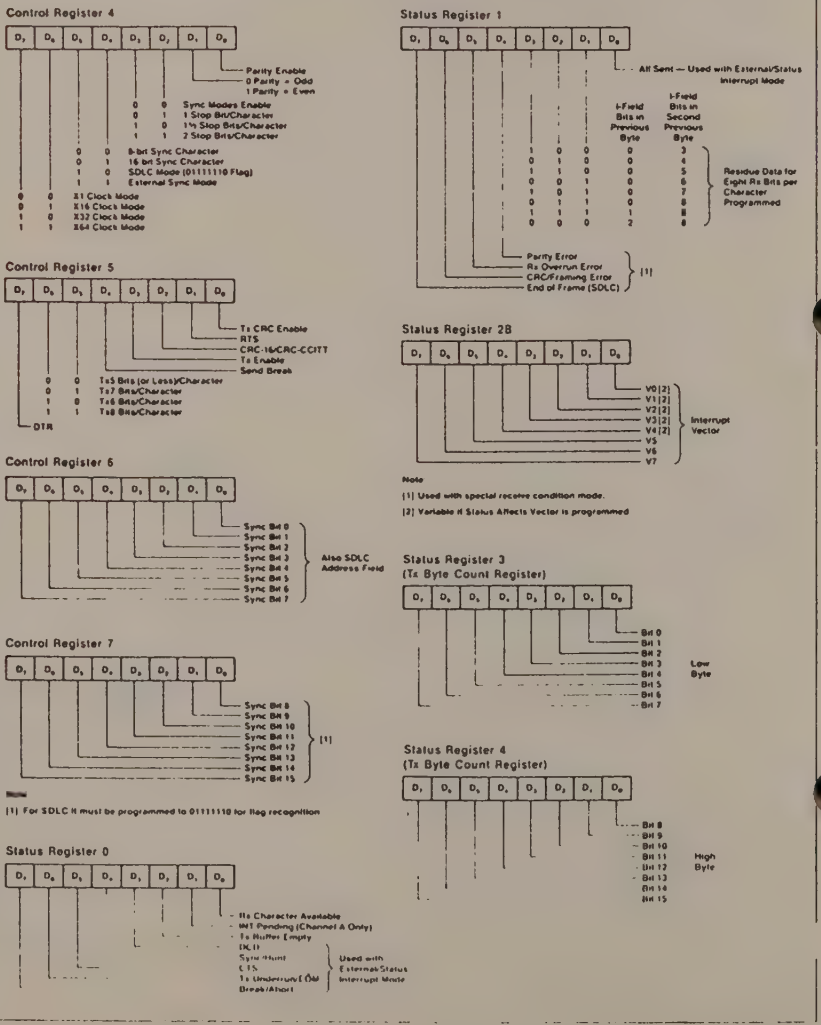
Table 13. Condition Affects Vector Modifications

Interrupt Pending (SR0, D_1 Channel A)	8085 Modes / 8086 Modes	D_6 D_7	D_3 D_1	D_2 D_0	Condition
0		1	1	1	No interrupt pending
1		0	0	0	Channel B transmitter buffer empty
1		0	0	1	Channel B external/status Change
1		0	1	0	Channel B received character available
1		0	1	1	Channel B special receive condition
1		1	0	0	Channel A transmitter buffer empty
1		1	0	1	Channel A external/status change
1		1	1	0	Channel A received Character available
1		1	1	1	Channel A special receive condition

Status Register Bit Functions (Sheet 1 of 2)



Status Register Bit Functions (Sheet 2 of 2)



Package Information

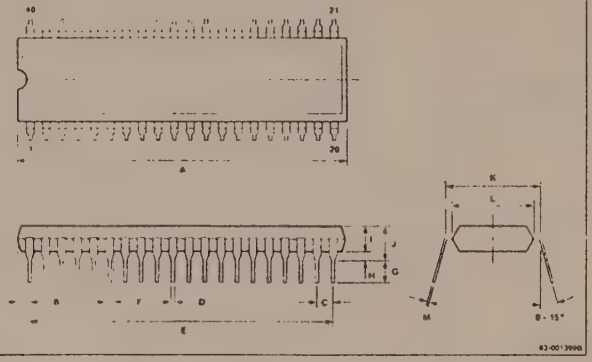
40-Pin Plastic DIP (600 mil)

Item	Millimeters	Inches
A	53.34 max	2.100 max
B	2.54 max	.100 max
C	2.54 [TP]	.100 [TP]
D	.50 ± .10	.020 +.004 −.005
E	48.26	1.900
F	1.2 min	.047 min
G	3.6 ± .3	.142 ± .012
H	.51 min	.020 min
I	4.31 max	.170 max
J	5.72 max	.225 max
K	15.24 [TP]	.600 [TP]
L	13.2	.520
M	.25 +.10 −.05	.010 +.004 −.003

Notes

[1] Each lead centerline is located within .25 mm [.010 inch] of its true position [TP] at maximum material condition.

[2] Item "K" to center of leads when formed parallel



83-001399B

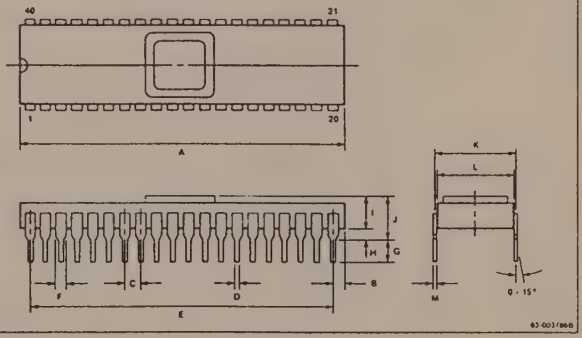
40-Pin Ceramic DIP (600 mil)

Item	Millimeters	Inches
A	53.34 max	2.100 max
B	2.54 max	.100 max
C	2.54 [TP]	.100 [TP]
D	.46 ±.05	.018 ± 002
E	48.26	1.900
F	.92 min	.036 min
G	3.5 ±.03	.0138 ±.012
H	1.0 min	.039 min
I	2.64	.104
J	4.57 max	.180 max
K	15.24 [TP]	.600 [TP]
L	14.93	.588
M	.25 ±.05	.010 +.002 −.003

Note:

[1] Each lead centerline is located within .25 mm [.001 inch] of its true position [TP] at maximum material condition.

[2] Item "K" to center of leads when formed parallel



83-001396B

Also available from NEC ELECTRONICS INC.
on the uPD7201A:

1. "General Systems Application Guide".

2. "Application Note 28: MPSCC Bit -
   Synchronous Data Communications".

3. "Application Note 31: Bisync Operation
   With The uPD7201A MPSCC".

NEC
NEC Electronics Inc.
CORPORATE HEADQUARTERS
401 Ellis Street
P.O. Box 7241
Mountain View, CA 94039
TLL 415 960 6000
TWX 910 379 6985

For Literature Call Toll Free: 1-800-632-3531
1-800-632-3532 (In California)

No part of this document may be copied or reproduced in any form or by any means without the prior written consent of NEC Electronics Inc. The information in this document is subject to change without notice. Devices sold by NEC Electronics Inc. are covered by the warranty and patent indemnification provisions appearing in NEC Electronics Inc. Terms and Conditions of Sale only. NEC Electronics Inc. makes no warranty, express, statutory, implied or by description, regarding the information set forth herein or regarding the freedom of the described devices from patent infringement. NEC Electronics Inc. makes no warranty of merchantability or fitness for any purpose. NEC Electronics Inc. assumes no responsibility for any errors that may appear in this document. NEC Electronics Inc. makes no commitment to update or to keep current the information contained in this document.

©1986 NEC Electronics Inc. /Printed in U.S.A.